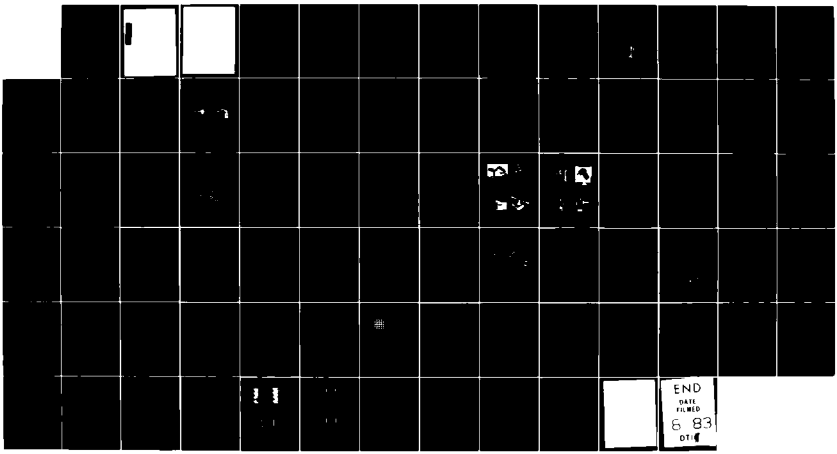
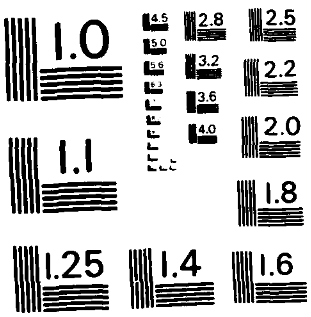
AD-A130 999 A SURVEY OF ROBOTIC TECHNOLOGY(U) DAVID W TAYLOR NAVAL SHIP RESEARCH AND DEVELOPMENT CENTER BETHESDA MD G CASTORE JUL 83 DTNSRDC-83/053 1/1

UNCLASSIFIED F/G 13/9 NL





MICROCOPY RESOLUTION TEST CHART
NATIONAL BUREAU OF STANDARDS - 1963 - A

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

REPORT DOCUMENTATION PAGE		READ INSTRUCTIONS BEFORE COMPLETING FORM
1. REPORT NUMBER DTNSRDC-83/053	2. GOVT ACCESSION NO. AD-A130 999	3. RECIPIENT'S CATALOG NUMBER
4. TITLE (and Subtitle) A SURVEY OF ROBOTIC TECHNOLOGY		5. TYPE OF REPORT & PERIOD COVERED 6. PERFORMING ORG. REPORT NUMBER
7. AUTHOR(s) Dr. Glen Castore		8. CONTRACT OR GRANT NUMBER(s)
9. PERFORMING ORGANIZATION NAME AND ADDRESS David Taylor Naval Ship Research and Development Center Bethesda, Maryland 20084		10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS Task Area Program Element 62760N Work Unit 1870-502
11. CONTROLLING OFFICE NAME AND ADDRESS Naval Air Systems Command (034C) Asst. Commander for Research and Technology Washington, D.C. 20361		12. REPORT DATE July 1983 13. NUMBER OF PAGES 79
14. MONITORING AGENCY NAME & ADDRESS(if different from Controlling Office)		15. SECURITY CLASS. (of this report) UNCLASSIFIED 15a. DECLASSIFICATION/DOWNGRADING SCHEDULE

16. DISTRIBUTION STATEMENT (of this Report)

APPROVED FOR PUBLIC RELEASE: DISTRIBUTION UNLIMITED

17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)

18. SUPPLEMENTARY NOTES

19. KEY WORDS (Continue on reverse side if necessary and identify by block number)

Robots
Robotic Technology

20. ABSTRACT (Continue on reverse side if necessary and identify by block number)

Robotic Technology is surveyed as a prelude to examination of its use in Naval Air Maintenance tasks. Topics include Robot Classification schemes, programming techniques, power systems, manipulators, control systems, sensors, and end effectors.

DD FORM 1 JAN 73 1473 EDITION OF 1 NOV 65 IS OBSOLETE
S/N 0102-LF-014-6601

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

TABLE OF CONTENTS

Page

LIST OF FIGURES

Page

Page

LIST OF TABLES

ABSTRACT

Robotic Technology is surveyed as a prelude to examination of its use in Naval Air Maintenance tasks. Topics include Robot Classification schemes, programming techniques, power systems, manipulators, control systems, sensors, and end effectors.

ADMINISTRATIVE INFORMATION

This report was sponsored by the Naval Air Systems Command under the Fleet Logistics Readiness Technology (AIR), Subproject Number WF 60-531, SFP 3314.

1. DEFINITION AND CLASSIFICATION

1.1 INTRODUCTION

The word "robot" has many definitions. Any study involving robotics must therefore make some attempt to define the term in order to specify the technologies being considered. The robots in use today are primarily machines with manipulators that can be easily programmed to do a variety of tasks automatically. After much deliberation the Robot Institute of America developed the following definition of a robot:

> A robot is a reprogrammable multifunctional manipulator designed to move material, parts, tools, or specialized devices through variable programmed motions for the performance of a variety of tasks.

Whatever its shortcomings, the spirit of the definition is clear. To be ranked as a robot a machine must have some type of built-in intelligence or programmable memory so it can operate automatically, and it must be capable of doing a fairly wide range of tasks.

For this project (applications of robotics to maintenance of naval aircraft) it has proved useful to develop a characterization which emphasizes the structural components of a robot. This emphasis makes the description compatible with the parametric representation used elsewhere in the project. In this survey then,

> A robot is a machine with three components: a multifunctional manipulator to move objects and tools; a controller to store data and direct the manipulator; a power system for the manipulator.

The manipulator consists of a combination of mechanical linkages and joints, drive motors, and feedback devices such as limit switches, resolvers, and tachometers. The manipulator is used to move objects or handle tools. In operational terms "multifunctional" means that the manipulator has three or more independent degrees of freedom. This flexibility allows it to reach any point in some volume of space and thus enables it to be used for a variety of tasks. Figure 1-1 is an illustration of the manipulator of the Unimation PUMA 600 robot. Notice that the manipulator includes the arm and wrist but not the hand or gripper. Manipulators are discussed in more detail in Section 4.

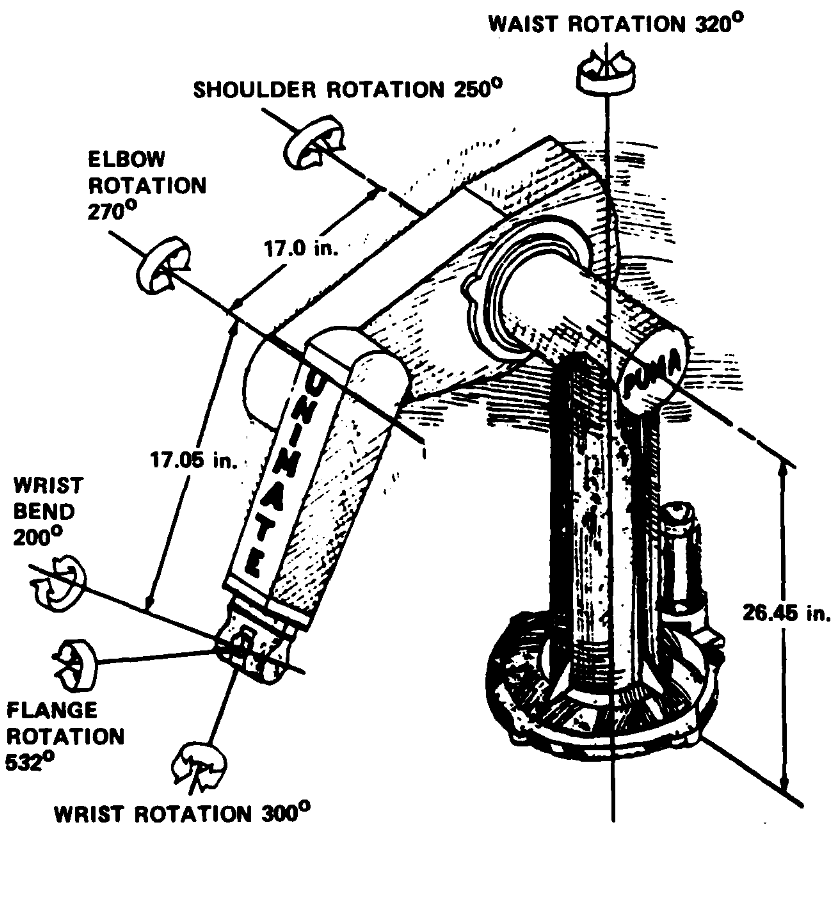


Figure 1-1 - PUMA 600 Manipulator

The controller functions as the brains of the robot. It stores data on the position of the manipulator and the sequence of motions and acts on data arriving from various sensors. Using these data it initiates and terminates the motions of the manipulator. The controller might be one or several microprocessors, a minicomputer, or something as mundane as a stepping drum. Control strategies are discussed in detail in Section 5. The diagram in Figure 1-2 illustrates some of the relations among the manipulator, the controller, and external sensors.[1*]

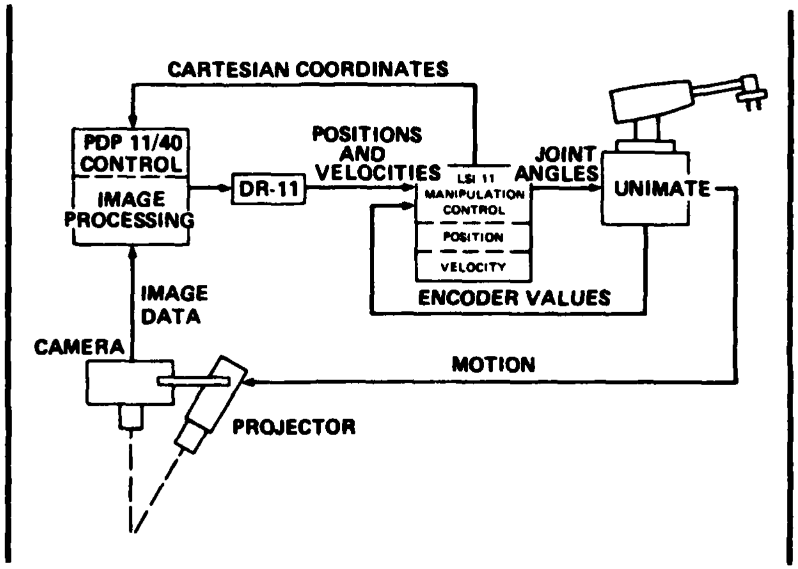


Figure 1-2 - Relationships between the Manipulator and the Controller

The power system, or drive system, supplies the force which enables the manipulator to do work. It is hydraulic, pneumatic, electric, or perhaps some combination of these. Typical electric drive units are D.C. servomotors and stepping motors. A hydraulic drive might use a simple piston or some type of hydrostatic motor. In any case practically all modern industrial robots require access to electric power for the controller. Drive systems are discussed in more detail in Section 3.

*A complete listing of references is given on page 69.

Equipment such as end effectors, which are the hands of the robot, specialized tools and jigs, and parts feeders may be required to make the robot useful. In this survey none of this equipment is treated as an intrinsic part of the robot, since it is generally not included when a robot is purchased and often must be specially constructed for the job at hand.

1.2 ROBOTS AND HARD AUTOMATION

The distinction one sometimes wishes to draw between robots and hard automation is usually one of degree rather than kind. In so far as there is a distinction, it probably rests on the extent to which software is used in machine operation. A robot is programmed to do a particular task rather than being physically constructed to do that task. Usually this means that a robot loses some precision as compared to a specially made machine, but it gains a tremendous amount in flexibility.

This contrast between robots and hard automation is illustrated by the following example. Grumman Aerospace Corp. has developed a machine to drill wing panels. This machine, called the *Automated Fixture Drilling System*,[2] consists of a unit with five degrees of freedom for holding and moving the panel, a controller, and a power system. However, the machine is designed so' y to drill large pieces of formed metal, and it can do very little else. General Dynamics has experimented with a robot to do the same job. This robot can be reassigned to a very different task such as welding, machine loading, or dye-casting.

1.3 ROBOT CLASSIFICATION

The purpose of classification schemes from the viewpoint of this project is to provide systematic ways to relate robot characteristics to characteristics of the jobs the robot is to do. Robots are classified in several ways. Classification by control mode emphasizes the way in which the manipulator is controlled. Work envelop classification is based on the shape of the region in space that the robot can reach. This shape is determined by the design of the manipulator joints and links. Finally, the Japanese classify robots according to teaching mode, i.e., the manner in which a robot is programmed to do a particular job. None of these classification schemes can be considered complete, since progress in designing robots has resulted in a good deal of overlap among once clearly distinguishable robot types.

1.3.1 Control Mode Classification

In this scheme robots are of two basic types: non-servo or servo. The servo robots are further divided into point-to-point and continuous path types. The essential difference between servo and non-servo robots is the presence of a feedback loop in the servo robots. This system consists of sensors, which read the actual position of the robot joints, and a program which compares these data with the programmed positions. The difference between the actual and desired position, coded as an error signal, is amplified and sent as a command signal to the valves for the actuator of each joint to correct the position of the manipulator. This classification method is summarized in Figure 1-3.

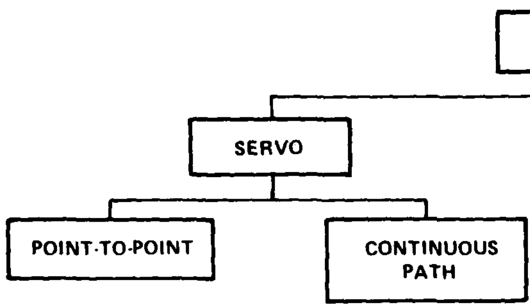


Figure 1-3 - Control Mode Classification

A non-servo robot is controlled by mechanical stops and limit switches. The following steps are typical of the operating sequence for a non-servo robot:[3]

(1) Upon start of program execution, the sequencer or controller initiates signals to control valves on the manipulator's actuators.

(2) The valves open, admitting air or oil to the actuators, and the members begin to move.

(3) Limit switches signal the end of travel to the controller, which then commands the controller valves to close.

(4) The sequencer then indexes to the next step and the controller again sends signals. These may be to the control valves on the actuators again or to an external device such as a gripper.

(5) The process is repeated until the entire sequence of steps has been executed.

With non-servo robots the arm can attain a relatively high speed because of the smaller size of the manipulator and the full flow of oil or air through the control valves. These robots are relatively low in cost, simple to operate and maintain, and are very reliable. They do have limited flexibility in terms of program capacity and positioning capability. Typically they are used as pick-and-place machines.

In a point-to-point servo robot the programmer specifies a fairly small number of points for the arm to reach and the order in which they are to be reached. For a 1973 Unimate 1000 series the maximum number of points the controller could handle was 500. In a 1981 Cincinnati Milacron HT-3 operating in point-to-point mode the number of points is 64,000. Such a robot is generally taught by moving the arm physically through the required sequence of motions and pushing a "record this point" button to record particular points in the trajectory. The controller is then put on automatic and the robot goes through the sequence itself. The path which the manipulator follows between the programmed points is not programmed and may differ from the path taken during the teaching sequence (see Figure 1-4).

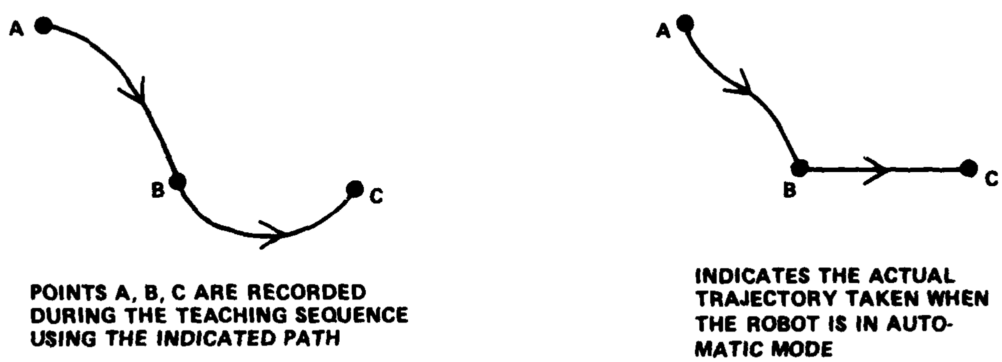


Figure 1-4 - Trajectory of a Point-to-Point Robot

Generally point-to-point servo controlled robots have high capability control systems with random access to multiple programs, subroutines, branches, etc. and thus provide great flexibility to the user. Hydraulic drives are most common, so these robots tend to lie at the upper end of the scale in terms of load capacity and working range. Most robots in place in the U.S. today (1981) are point-to-point servo-controlled.

In the servo-controlled continuous path robots the programmer specifies a path in space that the arm should take. Usually this is done by putting the controller in a record mode and moving the arm over the desired path either by hand or using some sort of joy stick. During programming and playback the sensor data on joint positions are sampled on a time basis rather than as discrete points in space. Typical sampling frequencies range from 60 Hz to 80 Hz, although frequencies as low as 4 Hz are sometimes used.

The rate of sampling enables many spatial positions to be stored in memory. Mass storage systems such as magnetic tape or magnetic discs are generally used. Some controller and data storage systems allow more than one program to be stored in memory and randomly accessed.

Continuous path robots are generally smaller than point-to-point robots. They are commonly used for spray painting, arc welding, polishing and grinding, and assembly. Some robots, such as the Cincinnati Milacron HT-3, are now manufactured with both continuous path and point-to-point control formats. In addition, many commercially available robots can be programmed off-line using a textual language similar to PASCAL or BASIC.

1.3.2 Work Envelope Classification

The work envelope of a robot is the region of space that can be reached by the end of its wrist. When a tool, such as a drill, is attached to the wrist, the effective working volume will be slightly different than the work envelope pictured here. Moreover, the end effector can reach some points within the work envelope only in certain orientations. The overall shape of the work envelope is determined by the structure of the manipulator joints and the lengths of the manipulator links.

There are three main work envelope types: Cartesian, cylindrical, and spherical. The illustrations in Figure 1-5[4] indicate the work envelopes corresponding to certain joint configurations. Note that there are two joint arrangements which give rise to a spherical work envelope. This classification is not all-encompassing, as it is possible to combine various joint types to achieve irregularly shaped work volumes.

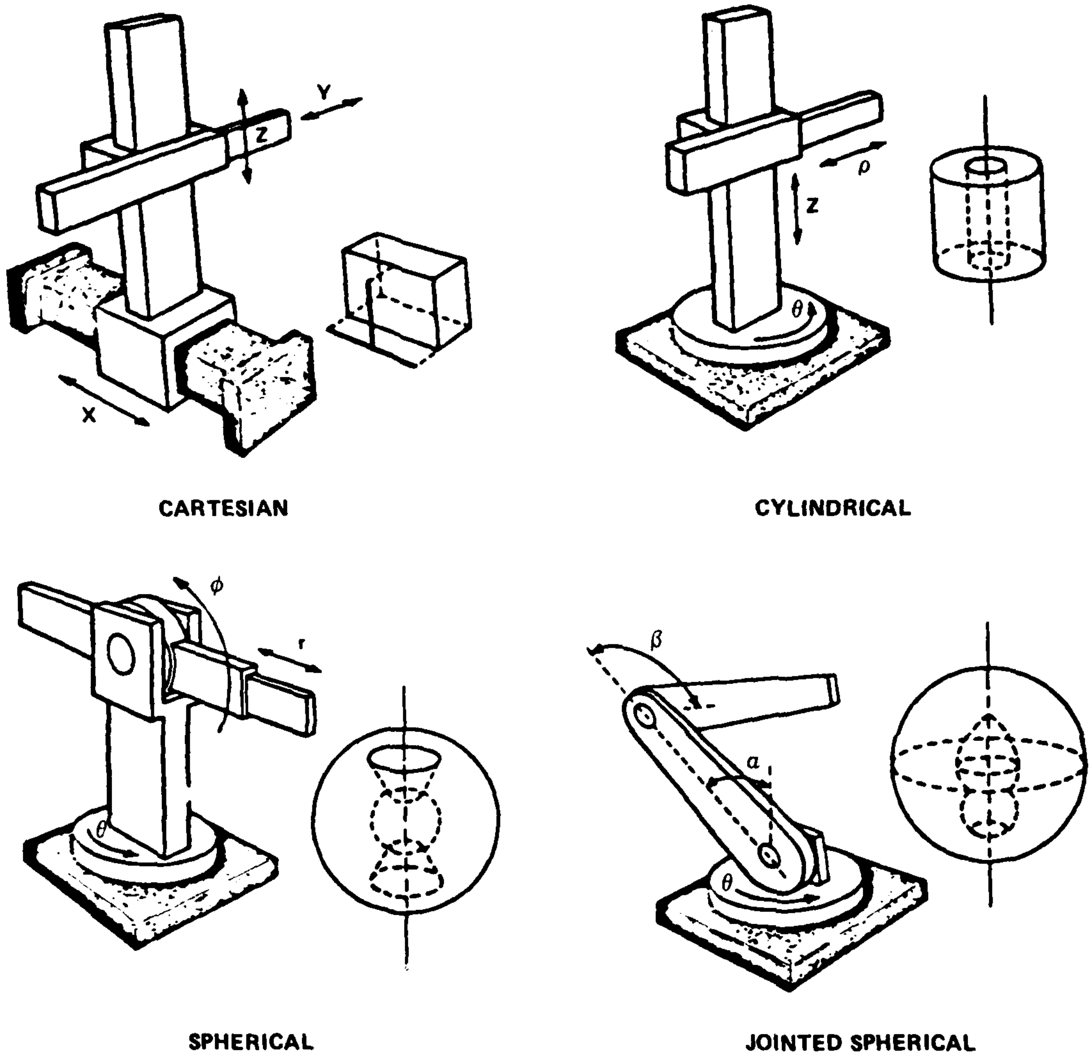
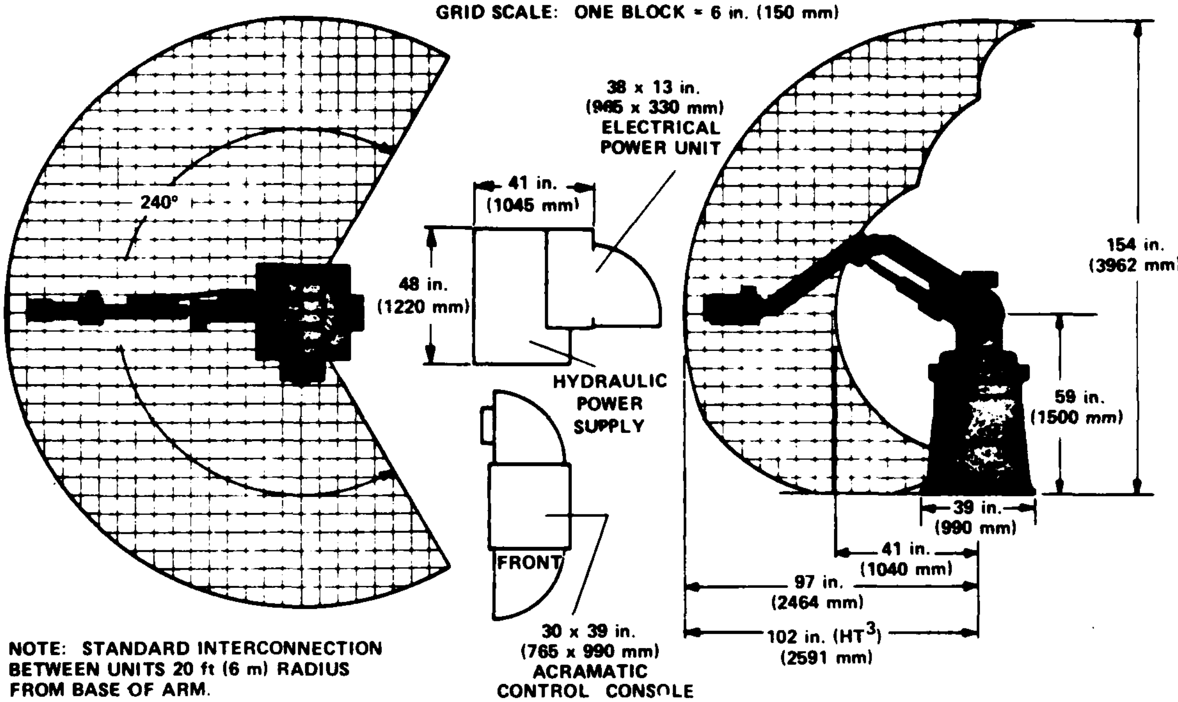


Figure 1-5 - Robot Work Envelopes and Joint Configurations

This classification by work envelope, or joint configuration, can be useful in matching robots to tasks. For example, the Cartesian robot in Figure 1-5 must move its entire mass during any x-axis translation. Therefore it will probably be very inefficient for jobs requiring fast left and right moves. On the other hand its motions are well adapted for handling wide flat sheets.

Spherical and cylindrical robots also have certain dynamic and kinematic advantages and disadvantages. For instance, the bulk of their mass usually lies near the first axis of rotation. Thus when the robot makes large horizontal translations, it spends little energy moving itself; the torque needed to move the workpiece is also less than that needed by the Cartesian robot. Such design advantages can save a lot of time and money.

In the literature supplied by manufacturers the work envelope is usually presented in the format shown in Figure 1-6.



Basic range and floor space drawings

Figure 1-6 - Cincinnati Milacron T-3 Working Volume

Since none of the joints of the T-3 rotate a full 360 degrees, the work envelope is only a portion of the sphere indicated in Figure 1-5.

1.3.3 Classification by Teaching Method

This is the standard Japanese classification for industrial robots.[5] It uses input information and teaching method to draw distinctions among robots.

In this classification scheme a "manipulator" is defined as

> ... a device for handling objects as desired without touching with the hands and it has more than two of the motion capabilities such as revolution, out-in, up-down, right-left traveling, swinging or bending, so that it can spatially transport an object by holding, adhering to, etc.

A "robot" is defined as

> ... a mechanical system which has flexible motion functions analogous to the motion functions of living organisms or combines such motion functions with intelligent functions, and which act in response to the human will. In this context intelligent functions mean the ability to perform at least one of the following judgment, recognition, adaptation, or learning.

The categories of industrial robots are as follows:

Manual manipulator	-	A manipulator that is directly operated by a person
Sequence robot	-	A manipulator, the working step of which operates sequentially in compliance with preset procedures, conditions, and positions.
Fixed sequence	-	A sequence robot defined as above, for which the preset information cannot be easily changed
Variable sequence	-	A sequence robot as defined above, for which the preset information can be easily changed.
Playback robot	-	A manipulator that can repeat any operation after being instructed by a person.
Numerically controlled robot	-	A manipulator that can execute the commanded operation; in compliance with the numerically loaded; information on e.g. position, sequence, and condition.

Intelligent robot - A robot that can determine its own actions through its sensing and recognition abilities.

This classification method meshes very well with the hierarchy of programming techniques established in Section 2.

The three classification methods presented here are useful tools in the attempt to develop systematic methods for relating robot characteristics to the requirements of aircraft maintenance tasks. As robotics technology matures, more sophisticated and flexible characterizations of the robot's skills will undoubtedly emerge.

2. PROGRAMMING ROBOTS

2.1 PROGRAMMING TECHNIQUES

The methods for programming a robot vary widely. There are "action" languages through which the robot is programmed by leading it through the task. Robots programmed using these "teach-by-show" methods fall into two categories. In the first, used primarily for point-to-point control systems, the programmer moves the arm to the desired points by hand or with a joystick and then presses a button on a recording pendant. The position of that point and its order in the sequence of moves is then recorded. When the robot is put into automatic mode, it will go to the recorded points in the order recorded.

In the second type of teach-by-show programming the user leads the arm through the desired path. The robot controller samples the position data at some fixed rate, say 80 Hz. These data are stored as a program. This programming method is generally used on continuous path robots and often includes some provisions for subroutines and velocity control.

Both of these teach-by-show methods are on-line programming methods, meaning that both the programmer and the robot are at the work station and the programmer leads the robot through the task. There are advantages and disadvantages to each of these types of programming. Because there is a one-to-one correspondence between program statements and robot actions, programs are fairly easy to debug. For simple jobs debugging is fairly quick and requires relatively unskilled personnel. On the other hand, the programs are very specific. Doing another similar task usually means completely reprogramming the robot. Doing the same task but with another robot also means constructing a new program. The programs are also machine specific. For assembly operations this type of programming becomes inefficient.

The latest generation of commercial robots can be programmed almost as one would program a computer. The program statements refer to end effector positions. There are specific statements for tool actions, such as closing a gripper, and provisions for subroutines based on simple sensor data. An example of such a program is provided in the next section. Such programs require a data base containing information, such as position and orientation of parts, about the job at hand. This information is often gathered by moving the end effector to the desired position and recording the point. In other words, the arm functions as an analog-to-digital recording device.

This type of programming is attractive for several reasons. It is straightforward to incorporate simple sensors into the control circuit. Motions can be based on the results of computations, which means the robot's environment need not be extremely well-ordered. Programs can be written off-line and "fine-tuned" when put onto a particular machine. But there are problems too with this type of programming. Debugging becomes much more difficult, since one movement may depend upon several computations and on reading from several sensors as well as on data originally stored in the memory. For much the same reasons, program generation becomes more difficult and requires more highly trained personnel.

The direction of much research in robot programming is toward goal-based programming. Such a program would contain only a description of the desired task, such as "build 50 number 6A drill with type 3 chucks". The robot control system then builds an assembly routine from a stored set of subroutines, parts data, and a "world model" which enables the system to interpret sensor data. First steps in this direction have been taken by several research groups. The various programming techniques are summarized in Figure 2-1.

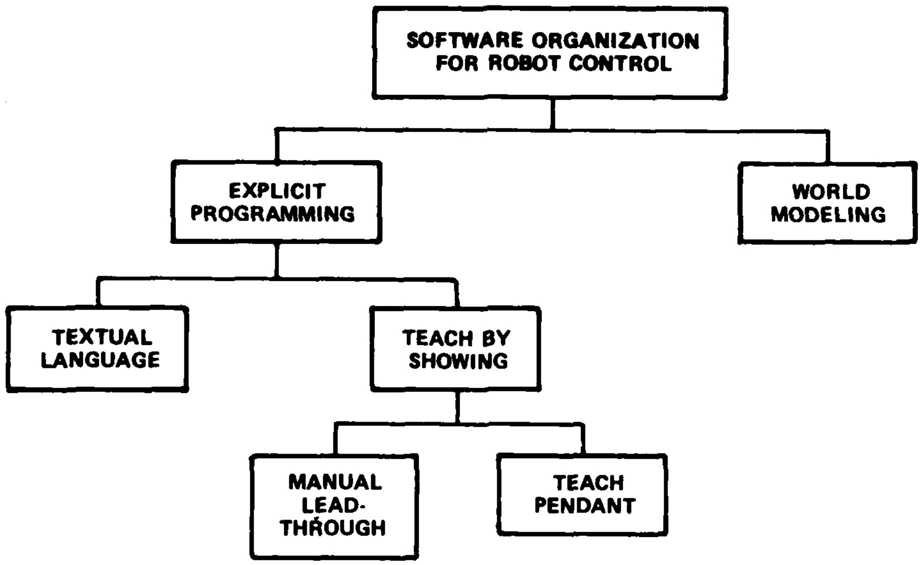


Figure 2-1 - Programming Techniques

In explicit programming the user is responsible for directing the specific motions of the manipulator. In world modeling systems the robot is given a very general instruction such as "assemble 5 copies of part #83". Specific motions are then determined by the robot control system.

Each teaching method works well in certain situations. Paint-spraying is best handled using manual lead-through techniques; pick-and-place operations can be programmed quickly using a teach pendant; assembly operations demand some type of textual programming.

The off-line programming languages can be divided into four levels, as shown in Table 2-1. The levels correspond roughly to the area in which the programmer focuses his attention in the manipulator-task relationship when programming.

TABLE 2-1 - OFF-LINE PROGRAMMING LANGUAGE LEVELS

LEVEL	CHARACTERIZATION	EXAMPLES
1	For each move the programmer must provide a great deal of information about individual actuators and joint angles.	ML - IBM
2	Motions are described in terms of end effector position in space.	VAL - Unimation SIGLA - Olivetti TEACH - Bendix RAIL - Automatix
3	Moves are described in terms of the positions and motions of the objects being manipulated.	LAMA - MIT AL - Stanford AUTOPASS - IBM RAPT - Univ of Edinburgh
4	The programmer needs only to give a clear description of the over-all goal.	

The first two levels are explicit textual languages; levels 3 and 4 involve world modeling techniques. The languages in level 2 permit detailed control over the manipulator actions with such direct commands as "OPEN", "MOVE", and "PICK". The more powerful languages in this category are WAVE, developed at Stanford, and MAPLE and EMILY, both developed at IBM. Level 2 languages have been in industrial use for several years.

In the level 3 languages motions are described in terms of the positions and motions of the objects being manipulated. These programming methods utilize "world models" which are symbolic representations of the manipulator, the workspace, and the objects involved in the task. Basically this is control via simulation. With these languages the user enters general task commands such as "PLACE INTERLOCK ON BRACKET SUCH THAT INTERLOCK HOLE IS ALIGNED WITH BRACKET HOLE." The program then refers to a database and geometric model to select grip points, approach paths, and the sequence of moves necessary to assemble the parts. These languages are still in the development stage.

Figure 2-2[6] shows the relationships among some existing robot programming languages as well as their relationships to some common computer languages.

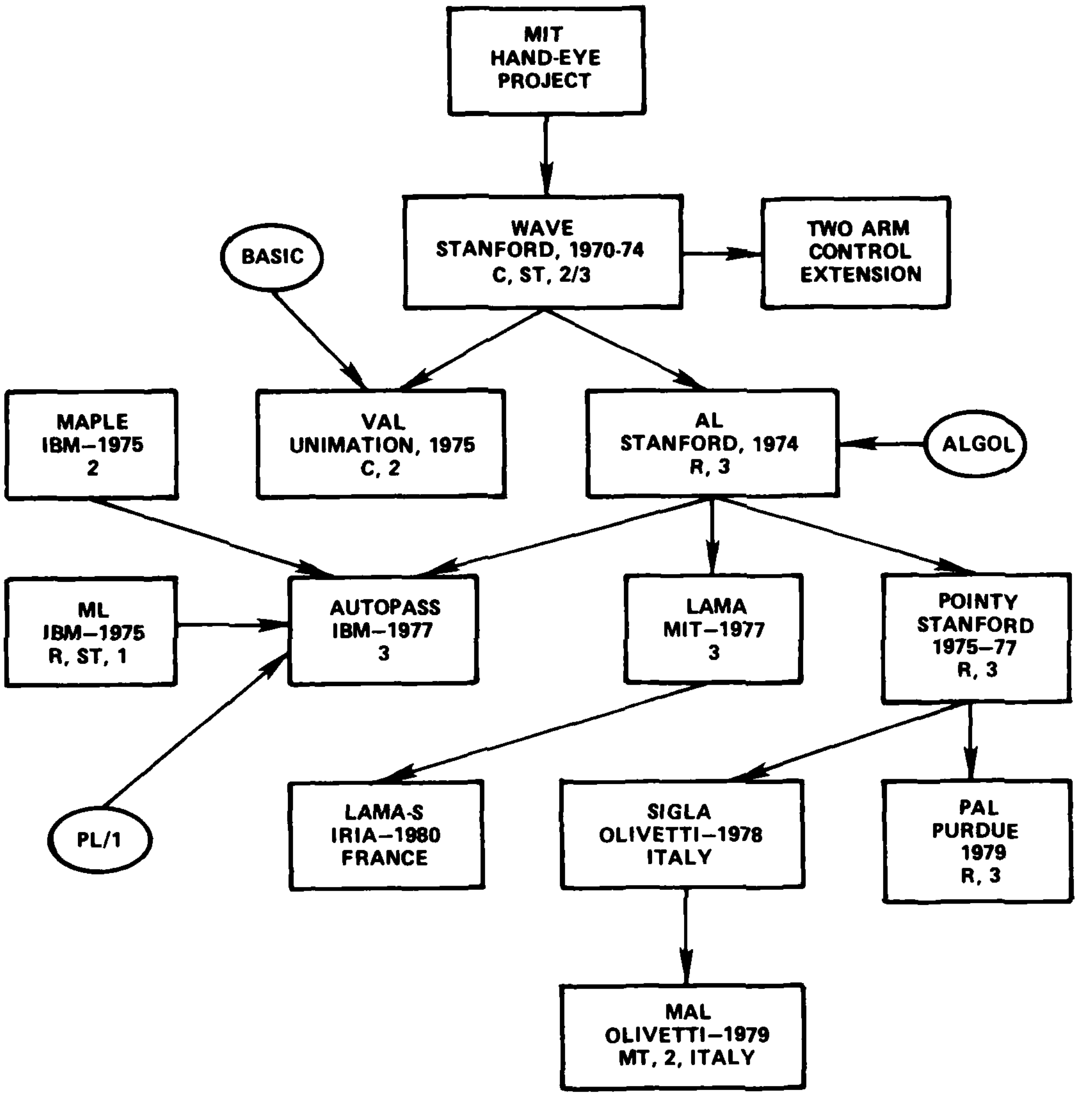


Figure 2-2 - Robot Programming Language Evolution and Classification

The summary information for each language includes the name, the date it first appeared in the literature, the developing organization, whether it is intended mainly for commercial (C) or research (R) use, whether it is considered a single task (ST) or multitask (MT) language, and the level of the language (1,2,3, or 4).

2.2 PROBLEMS WITH SOFTWARE

Current robot programming languages have two major shortcomings. First, they do not incorporate control features in a form which is directly applicable to machine control. Functions such as starting motors or actuating air cylinders cannot be expressed directly in the programming language. Instead, subroutines must be written by a programmer for each application. Ideally a language should include functions which allow rapid interaction with other aspects of the production or maintenance process. These functions range from coordinating operations of several other machines to monitoring and reporting data on production rates and piece quality.

The second problem with current languages is that programs are not transferrable from one machine to another. This difficulty manifests itself in two forms. One is that a program written for one robot cannot be immediately transferred to another robot of the same model. Calibration adjustments must be made because of differences in gear play, motor resistance and friction in electric drives, or leakage rates in hydraulic drive systems.

The second type of difficulty is that programs are usually specific to a particular joint configuration. Replacing a robot with a cylindrical joint configuration with one with a spherical joint structure requires rewriting the program. This problem may be important in Navy maintenance applications if attempts are made to develop standard programs for certain maintenance actions on, for example, avionics equipment or certain engine components.

2.3 STRUCTURE OF A LEVEL 2 LANGUAGE

The chart in Figure 2-3 summarizes the type of information contained in a second level language such as VAL.[7] A section of a VAL program is shown in Figure 2-4.[8] The program instructs the robot to pick up a metal plate and place it on a block.

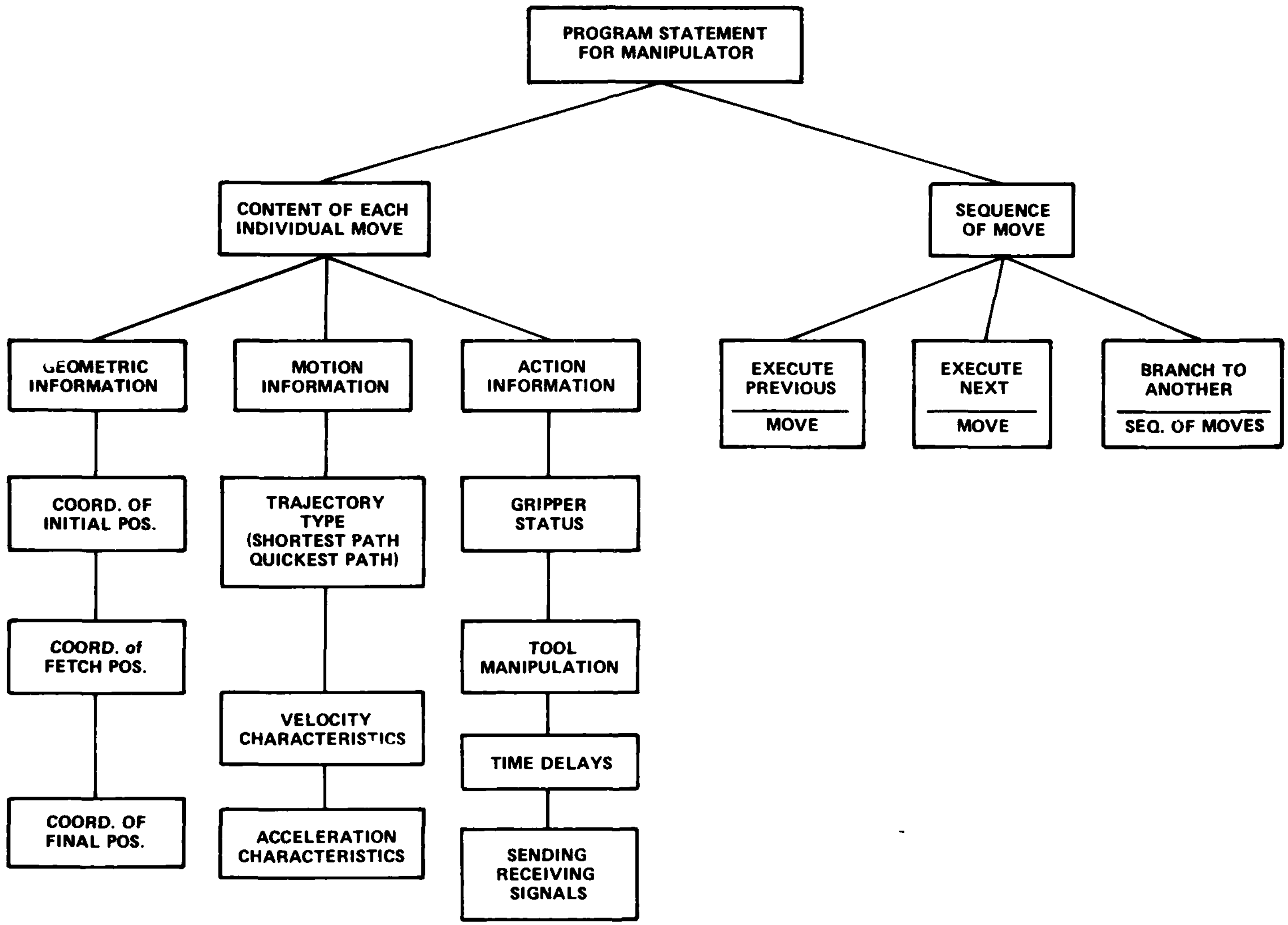


Figure 2-3 - Content of a Level 2 Robot Control Language

```
1. MOVE P1
2. MOVE P2
3. SPEED 30.00
4. MOVES P3
5. CLOSEI 0.00
6. MOVE P2
7. MOVE P1
8. SPEED 30.00
9. MOVES P4
10. SPEED 10.00
11. MOVES P5
12. OPENI 0.00
13. MOVE P6
14. RETURN 0.
```

Figure 2-4 - Sample VAL Program

The command "MOVE P1" means: move the end effector to point P1 in a straight line. "MOVES P3" means: move to P3 using joint interpolated motion. The program must include a data base containing the coordinates of the points P1, P2, etc.. These coordinates are usually recorded by using the arm itself as a digitizer, moving the end of the arm to the desired point, and pressing a "record" button on a teach pendant. Storing the data as joint angles means the program becomes machine-specific. Another robot will reach P1 from the same starting position through a different manipulator configuration. Storing position data in Cartesian coordinates means the robot must calculate the joint coordinates before it can act. The calculation is complex and requires a sophisticated algorithm and some computing power in the controller.

2.4 SUMMARY

The programming of a robot takes place at several levels which can be thought of hierarchically as

job	a set of tasks
task	a set of routines

routine	a set of subroutines
subroutines	commands to specific actuators, computations based on sensor data, etc.

For instance, the job might be to assemble an automobile alternator. A particular task could be to fasten on the faceplate. In a level 2 control language this task would be handled by a set of statements similar to the VAL program given in Figure 2-4. A routine is exemplified by a single command such as "MOVE" or "GRASP". In a level 2 language most statements specify routines. In a level 3 language the control system determines the routines. The subroutines involve direct control of the robot actuators through statements such as "VALVE(3,ON)". Program statements in a level 1 language are of this sort. The level of sophistication and type of knowledge required of the programmer changes considerably as one moves through this hierarchy of programming levels.

Arrangement of control languages into these levels in some sense captures the ease with which the robot can be programmed. It is faster and requires less detailed knowledge of the robot to use a level 3 language, with tasks as program statements, than to use a level 1 language in which joint angles, velocities, etc. must be specified for every motion.

In addition the ease of programming a language can be evaluated for the power with which the robot can be programmed The basic program functions can be viewed as tool commands, manipulator motions, sensor data processing, communications with other machines, decision making capability, and computation. A minimal language allows the operator to specify manipulator movement and tool commands. In more powerful languages the robot actions will depend on quantities calculated from extensive arrays of incoming signals reflecting the current state of the workplace and on internal variables reflecting the results of previously executed moves.

3. POWER SYSTEMS

Three basic types of power systems are used in robots today: electric, hydraulic, and pneumatic drives. This section discusses the principles underlying each type of system, some of the advantages and disadvantages of each type, and examples of commercial robots using each type of drive.

3.1 ELECTRIC DRIVE

There are three basic types of electric motors: direct current (D.C.), alternating current (A.C.), and stepping motors. A simplified picture of a D.C. motor is given in Figure 3-1.

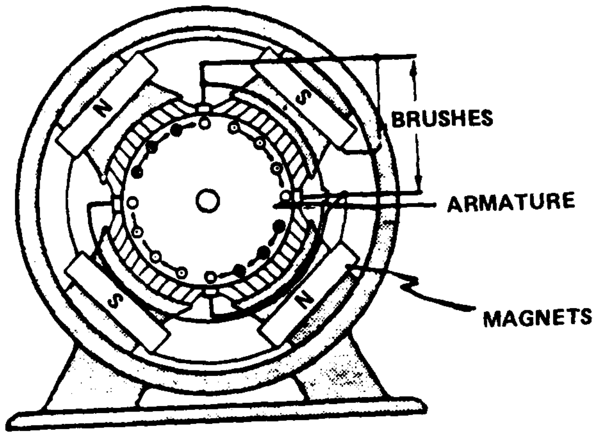


Figure 3-1 - D.C. Motor

In a robot the armature shaft of a motor is generally connected to some gears or ball screws to drive one of the links of the manipulator. The motion of that link is controlled by varying the current or voltage sent to the motor. Most U.S. manufacturers control the motion of the manipulator by altering the motor current. European and Japanese robot manufacturers generally control the motor voltage. As will be discussed in more detail in Section 5, torque control is best achieved through current manipulation, and position control is more readily handled by changing the motor voltage. The act of assembling two pieces usually requires torque control; the process of picking, moving, and positioning pieces for assembly requires position control. However, simultaneous control of both current and voltage is difficult and poses problems in control system design when a robot is intended for use over a wide range of tasks.

Each type of electric motor has certain advantages and disadvantages. D.C. motors start and stop quickly, have low inertia, and can run directly from batteries, but their complexity leads to maintenance problems. A complete actuator system generally requires gearing and/or ball screws which are subject to localized wear in the gears and corrosion on the ball screws and hence lead to maintenance problems. Brushless D.C. motors are used in applications such as paint spraying where sparking cannot be tolerated.

A.C. servo motors have a number of disadvantages when compared to D.C. motors and at present are rarely used in robots. Disadvantages center on two aspects of A.C. motor structure. First, the controls must be A.C. operated since the motor depends on a variable frequency or variable A.C. voltage. Second, since the rotor-stator structure acts as a transformer, it does not work well at low shaft speeds. Moreover, the starting and stopping procedure is more difficult than with a D.C. motor, since the motor operation depends on inducing a current in the rotor. Sometimes it is necessary to have a secondary, or starter, system to initiate rotor motion.

Despite these disadvantages, variable frequency A.C. motors have some significant advantages, such as accurate speed regulation and high energy efficiency, that suggest they would be useful in robotics. General Electric and Gould Inc. are developing A.C. servo technology to a level at which it could be incorporated into a robot drive system.

Stepping motors operate by electrical pulses. Each pulse advances or reverses the shaft a small but specific amount (say .002in.). Since they operate by means of pulses, these motors are easy to interface with digital controls. On the other hand there is a definite lower limit to the positional accuracy that can be achieved.

Stepping motors, A.C. motors, and D.C. motors differ from each other in fundamental performance aspects. The D.C. motors are the easiest to incorporate into servo systems, which accounts for the extent of their use in present day robots. G.E. is expected to develop robots with A.C. drives in the near future.

Table 3-1[9] gives several examples of electrically driven robots. The price given is the average price for a system installation.

TABLE 3-1 - ELECTRICALLY DRIVEN ROBOT CHARACTERISTICS

Model	Average price ($)	Payload (lb)	Accuracy (in.)	Repeatibility (in.)
Bendix ML-360	100,000	150	0.020	0.005
Bendix AA-160	70,000	45	0.004	0.002
G.E. A12 Allegro	125,000	14.3	0.0063	0.001
ASEA IRb-60	75,000	135	0.016	0.006

3.2 HYDRAULIC SYSTEMS

Hydraulic actuator systems consist of a fluid (usually oil) reservoir, a pump, and the motor or cylinder-piston units which actually drive the manipulator links. If the robot is servo-controlled, the actuator system will generally contain some type of electro-hydraulic servo valves. The relationships among these components are illustrated in Figure 3-2.[10]

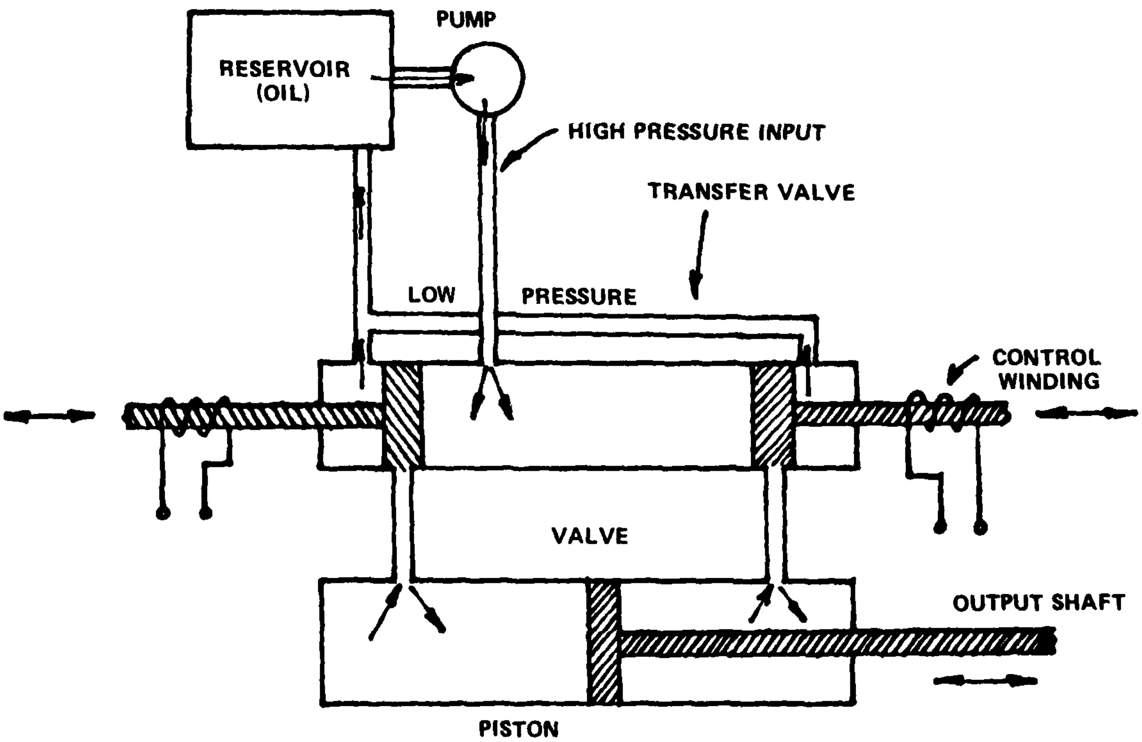


Figure 3-2 - Hydraulic Actuator System

The fluid reservoir should be large enough to supply the system pump for at least several minutes without any return flow. This reserve is required in order to (1) fill the system without exposing the filter and strainer, (2) maintain a stable oil level despite normal fluctuations in flow, (3) pump enough fluid to sustain the system while the parts coast to a stop during emergency shutdown. The size of the reservoir tank as well as the need for heat dissipation will influence the location aboard ship for use of a heavy duty robot such as the HT-3.

The hydraulic fluid itself is important. To minimize fire hazard phosphate-ester or water-glycol based fluids might be used. However, the use of water based fluids leads to maintenance problems because of increased corrosion.

In addition to the simplest hydraulic actuators, which are simple pistons, there are hydrostatic motors. The type most often used in actuators for robots is the gerotor-type motor (Figure 3-3).[11]

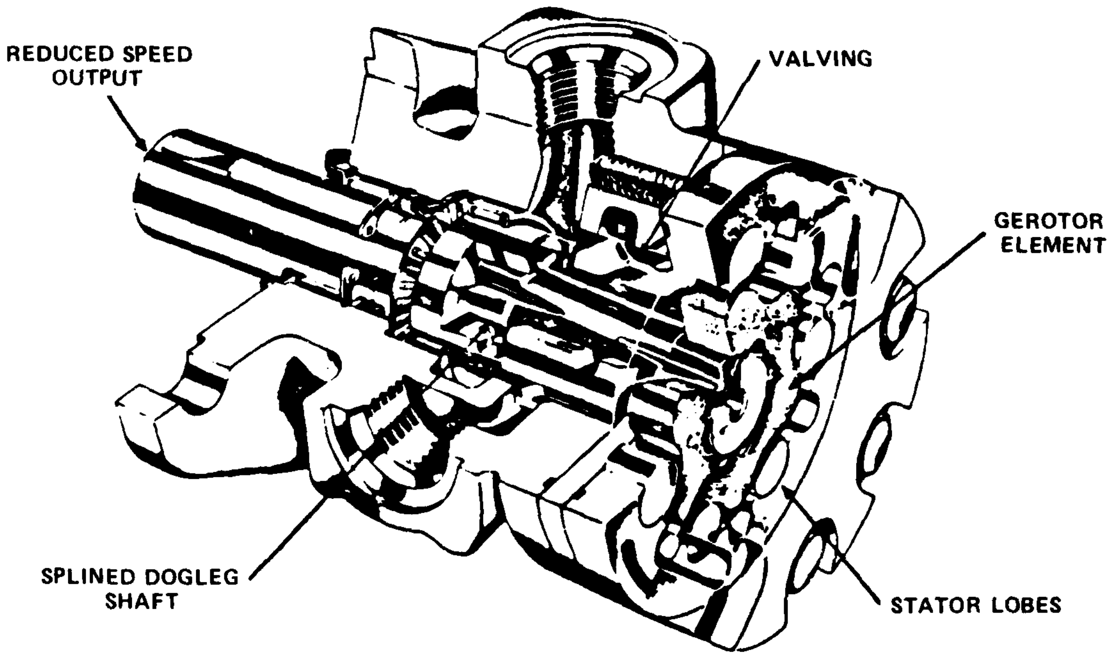


Figure 3-3 - Gerotor-Type Hydrostatic Motor

The gerotor design has the advantage of inherent torque multiplication because of nutation of its rotor, which has one less tooth than the stator. These motors serve as low speed, high torque drives and operate typically from 5 rpm to 1500 rpm, depending on displacement, and at pressures up to 3000 psi.

The types of tasks the robot is expected to do will, in part, determine the type of hydrostatic motor used. The significant aspects of the application are those which determine the power, torque, position, and velocity control that the robot must have. The flow into a motor determines the speed or angular velocity of the shaft; torque is controlled through the pressure. By analogy with the electrical motor, it can be seen that simultaneous control of position and torque is difficult.

Hydraulic robots have several problems. There is always leakage of the hydraulic fluid and the possibility of a hose breaking. Most servo-controlled hydraulic robots require electric power both for the pump and for the servo valves which are generally electro-hydraulic.

These robots also have some significant advantages over electrically driven machines. They have very few moving parts — only seven for a typical six-axis hydraulic robot -- which means fewer maintenance problems. They have a good weight-to-load ratio (compared to electric robots) which means that robots of reasonable size can handle heavy loads. The hydraulically powered Cybotech H80 robot, for instance, weighs 3900 lb and has a load capacity of 175 lb, a weight to load ratio of about 22:1. The electrically driven Bendix ML-360 weighs 6000 lb and has a maximum load capacity of 150 lb, giving it a weight to load ratio of 40:1. Some characteristics of other hydraulically driven robots are provided in Table 3-2.[9]

Fluid drive can deliver high power directly where it is needed. This factor, with the inherent compactness of a hydraulic actuator, minimizes the weight of the robot arm and thus increases payload. Robots designed to handle heavy loads (150 lb and more) are generally hydraulically actuated.

TABLE 3-2 - HYDRAULICALLY DRIVEN ROBOT CHARACTERISTICS

Model	Average price	Load capacity (lb)	Accuracy (in)	Repeatability (in)
Cincinnati Milacron T3-586	85,000	225	n/s	0.050
Cybotech H80	162,000	175	0.020	0.008
Thermwood Series 6	53,000	18	0.125	0.125
Unimate 4000	69,000	450	n/s	0.080

3.3 PNEUMATIC DRIVE

Pneumatic and hydraulic drives are based on similar principles. Since air is much less viscous than oil, valve tolerances must be tighter in pneumatically driven robots than in hydraulically driven ones to prevent leaks. There will be some leakage in any case, so a constant supply of air under pressure is needed to make up for the air lost through leakage. In addition, a filtering and drying system is required. Table 3-3 provides some idea of properties of pneumatically driven robots.[9]

TABLE 3-3 - PNEUMATICALLY DRIVEN ROBOT CHARACTERISTICS

Model	Average price ($)	Load capacity (lb)	Accuracy (in)	Repeatability (in)
Copperweld CR-100	45,000	11	0.0025	0.002
International Robotics IRI M50	12,500	50	n/a	0.040
Seiko Model 700	9,600	2.2	n/a	0.001

3.4 SUMMARY

Some broad comparisons can be made among these types of power systems. The pneumatic systems are generally much less expensive but, on the average, they have the smallest load capacity. Although the repeatability of pneumatic robots has improved significantly in recent years, most still do not offer velocity control (the IRI M50 does).

The electric machines, usually the most expensive, also achieve the best repeatability. Even the large Bendix ML-360 has a repeatability better than most of the hydraulically driven robots. On the other hand, the drive systems of electrical robots are intrinsically more complicated than those of either hydraulic or pneumatic systems and involve more maintenance problems.

4. MANIPULATORS

A manipulator consists of a collection of gears, drive motors, and joint assemblies usually housed within a metal shell. Figures 4-1[12] and 4-2[13] give some idea of the inner workings of these machines. In current commercial manipulators the individual links are built to be very rigid and the joints to have very little compliance, largely because flexibility in joints and links greatly complicates the control procedure. Achieving this stiffness arms requires quite massive arms. On the Cincinnati Milacron T3-576, for instance, the arm weighs 2000 lb and can lift 225 lb. The entire robot weighs 5000 lbs.

In Section 1 a classification of manipulators by work envelope and joint type was given (Figure 1-5). The four basic joint arrangements -- Cartesian, cylindrical, spherical, and jointed spherical -- generate characteristic work envelope geometries. To give a better idea of the range of shapes and sizes of commercial manipulators falling within each category, some illustrations are given in Figure 4-3.[9] The format for the captions is: power type; weight of robot; load capacity; joint type; and name.

According to Robert Cannon at Stanford University research is being done there on a very different type of manipulator. In this experimental design the manipulator links are allowed to be extremely flexible. Present work involves the problem of controlling one very flexible link moving in one dimension.

The manipulator link itself is made from two pieces of aluminum, each approximately 36"x2"x1/8", Figure 4.4. The problem is to move the end of the link from point A to point B in a very short time by rotating the control joint. While the end is stationary at B, the control joint continues to oscillate to "unwind" the first six harmonics of the wave set up in the link. The graphs in Figure 4-5 give some indication of the relative motion of the end of the link and the control joint. The next step in this experiment is to attach this flexible link to a PUMA 250.

At present a computational procedure for determining the shape of the work envelope from the arm geometry has not been fully developed. The most thorough

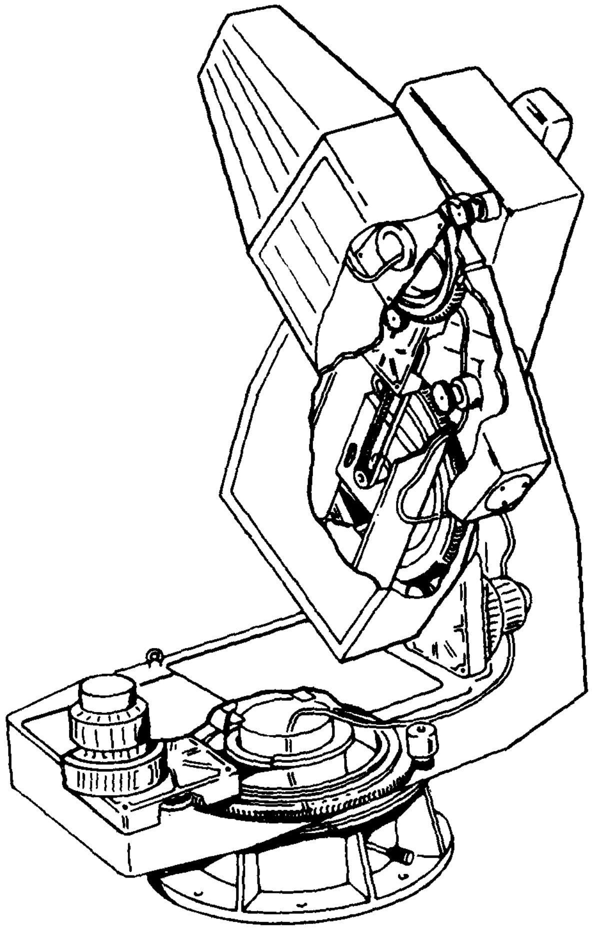


Figure 4-1 - Jungheinrich R100 (Electrically Powered)

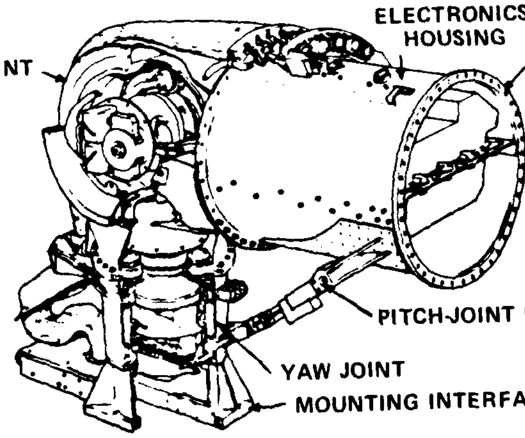
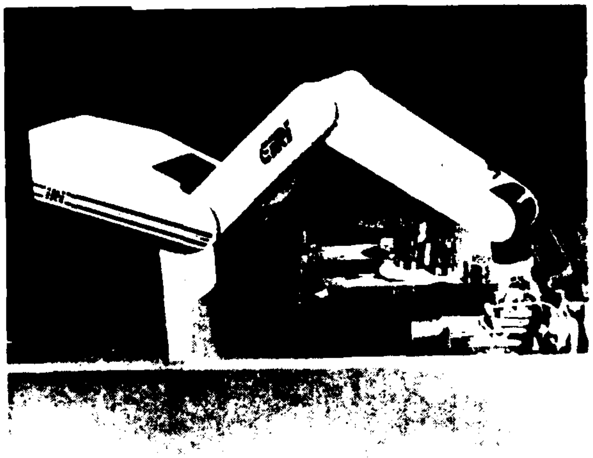
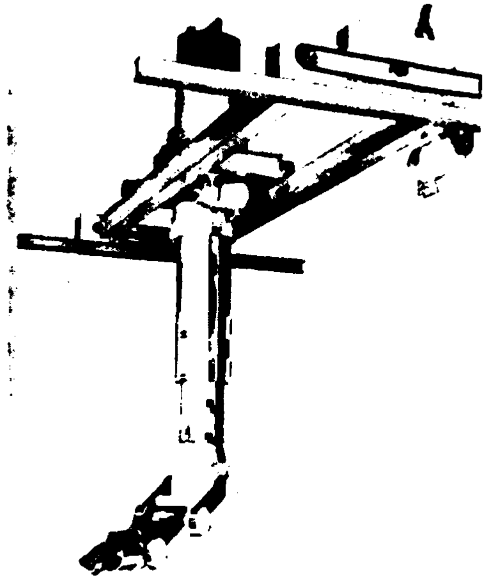


Figure 4-2 - Canadarm Shoulder

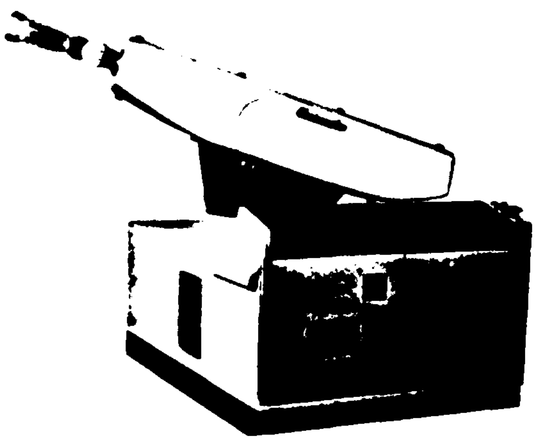
Figure 4-3 - Examples of Commercial Manipulators



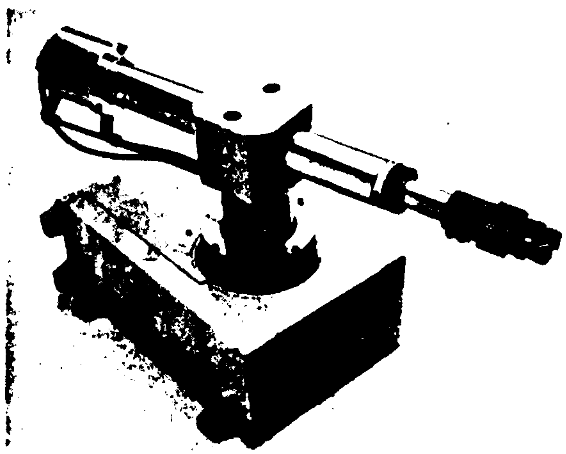
PNEUMATIC
200 lb
50 lb
JOINTED SPHERICAL
IRI-M50



ELECTRIC
2500 lb
500 lb Y-AXIS/100 lb JAWA
CARTESIAN/SPHERICAL
PAR XR-6100

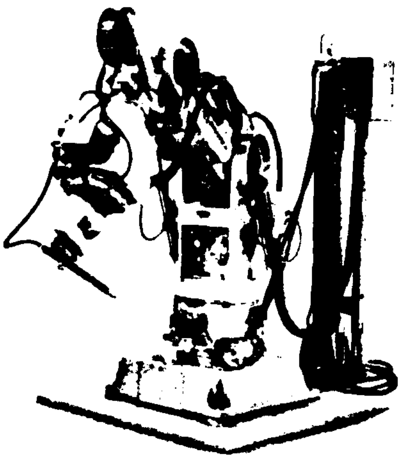


HYDRAULIC
3600 lb
300 lb
SPHERICAL
UNIMATE 2000

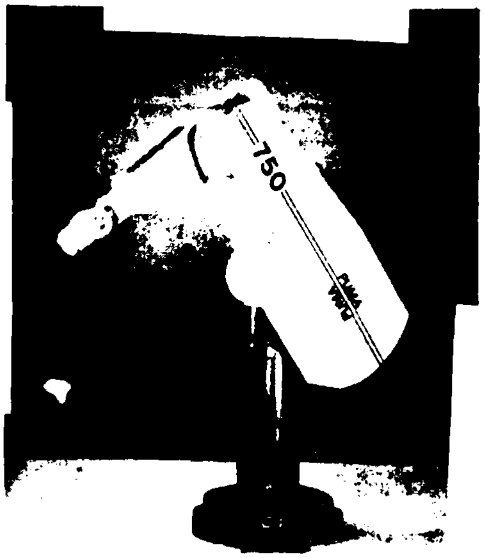


PNEUMATIC
44 lb
2.2 lb
CYLINDRICAL
SEIKO MODEL 700

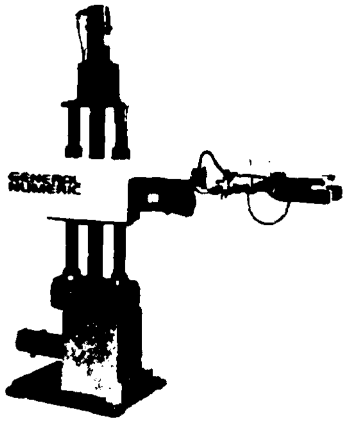
Figure 4-3 (Continued)



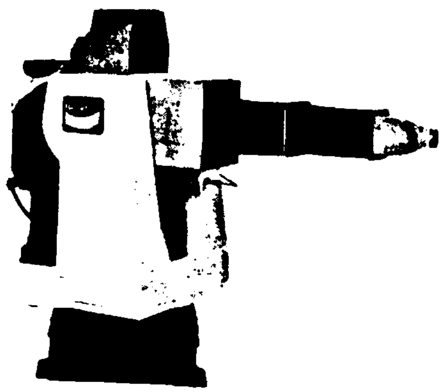
HYDRAULIC
3900 lb
175 lb
JOINTED SPHERICAL
CYBOTECH V80



ELECTRIC
N/S
N/S
JOINTED SPHERICAL
UNIMATE PUMA 750



ELECTRIC
1370 lb
44 lb
GENERAL NUMBER
SERIES M MODEL 1



ELECTRIC
3000 lb
45 lb
SPHERICAL
BENDIX AA-160

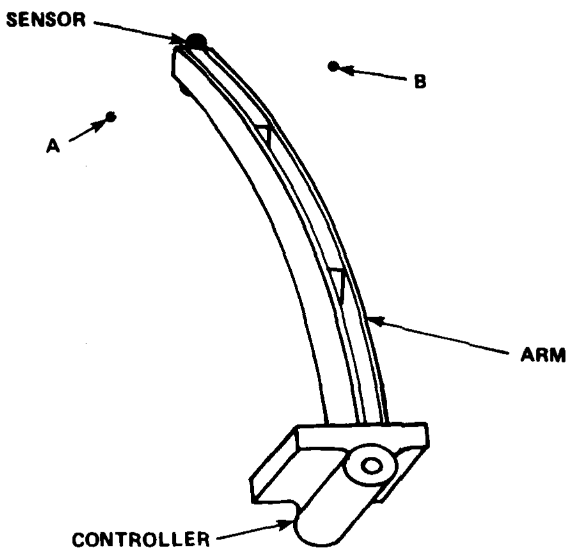


Figure 4-4 - Experimental Stanford Manipulator

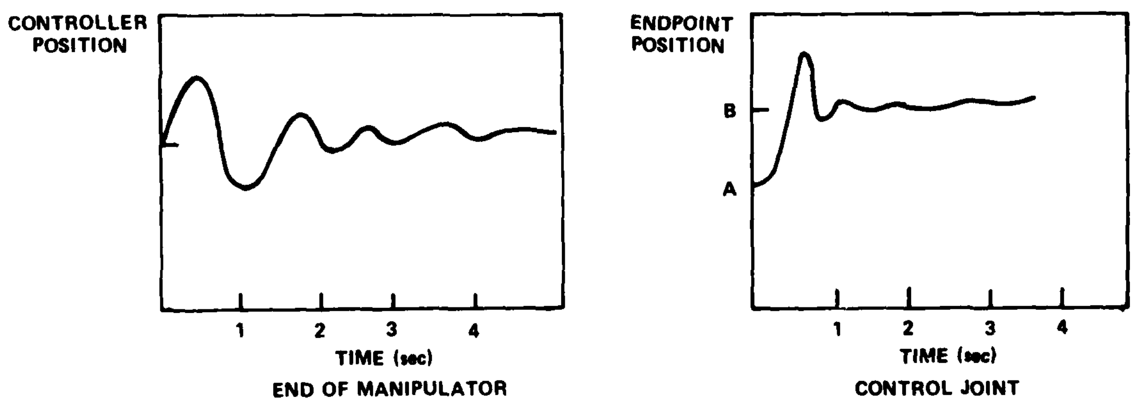


Figure 4-5 - Dynamic Behavior of Flexible Manipulator work in this direction has been done by Tsai and Soni at Oklahoma State University.[15] Studying two- and three-link manipulators moving in a plane, they developed explicit methods and formulae for determining the working volume. The crucial quantities in determining the shape of the workspace are (see Figure 4-6)[14] the ratio ℓ_2/ℓ_1, the difference $(\theta_{1max}-\theta_{1min})$ and θ_2.

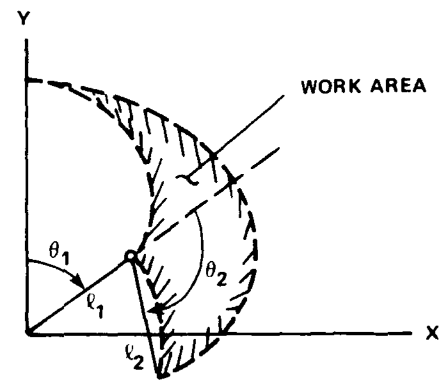


Figure 4-6 - Working Area Two-Link Manipulator

Important quantities for understanding and characterizing the behavior of manipulators are spatial resolution, control resolution, mechanical resolution, accuracy, repeatibility, structural frequency, and compliance. Spatial resolution refers to the smallest length or angle through which the tool tip can be guided. For instance, if a sliding joint is 36 inches long and the control unit has 8 bits dedicated to the control of that joint, then the smallest increment of motion is 36/256 or about 0.1". The resolution is reduced further by such things as slack in gears and corrosion and wear of parts. Spatial, mechanical, and control resolution are related in the following way: (see Figure 4-7).

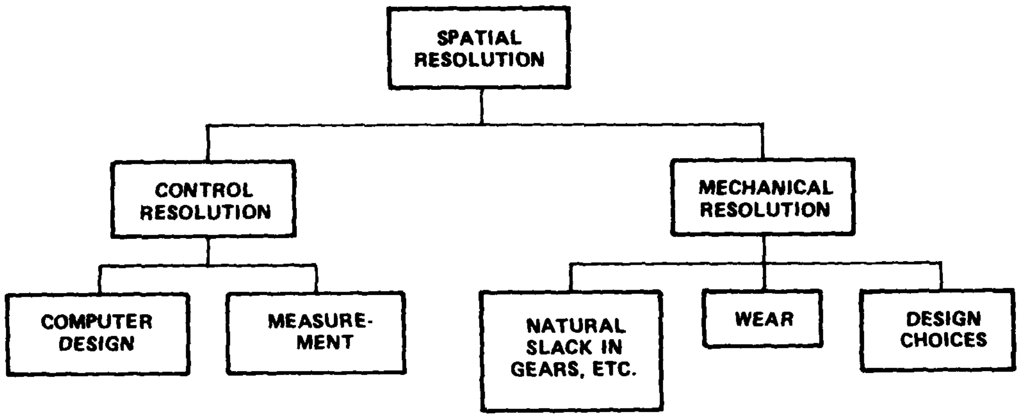


Figure 4-7 - Factors Influencing Spatial Resolution

Control resolution refers to limitations on the spatial resolution resulting from design choices within the computer control system or to limitations on the resolution of measurement devices such as resolvers, encoders, or potentiometers within the feedback loops. The example of 8 bits dedicated to the control of a 36-inch sliding joint is an instance of a computer design choice which affects spatial resolution.

The spatial resolution allowed by the control system design is eroded by aspects of the mechanical workings of the robot. There is a natural slack in gears and linkages which introduces inherent inaccuracies. Moreover, as the robot works, gears and bearings will wear and corrode in asymmetric patterns, leading to reduced resolution. Also, certain choices of design of mechanical subsystems affect resolution. Examples of these choices range from the number of teeth on a gear to the angular resolution of a stepping motor.

If a robot with a Cartesian configuration has a resolution of 0.1" for each of its three sliding joints, its working volume is effectively uniformly divided into small cubes 0.1" on a side. For revolute joints, however, the situation is quite different. As one moves farther away from the revolute joint, the volume of the "unresolved" region becomes larger. This effect is illustrated in Figures 4-8 and 4-9. In practice this effect is important because it means that, for robots with revolute joints, delicate motions learned in one part of the work volume cannot necessarily be transferred to other parts of the working volume.

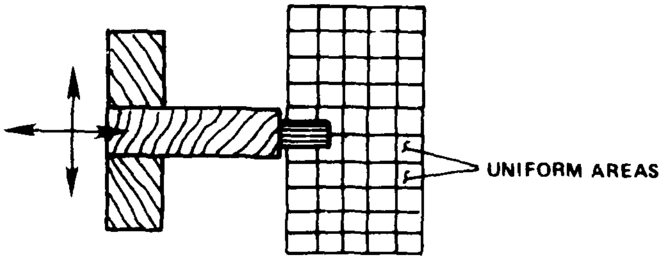


Figure 4-8 - Spatial Resolution of Cartesian Robot

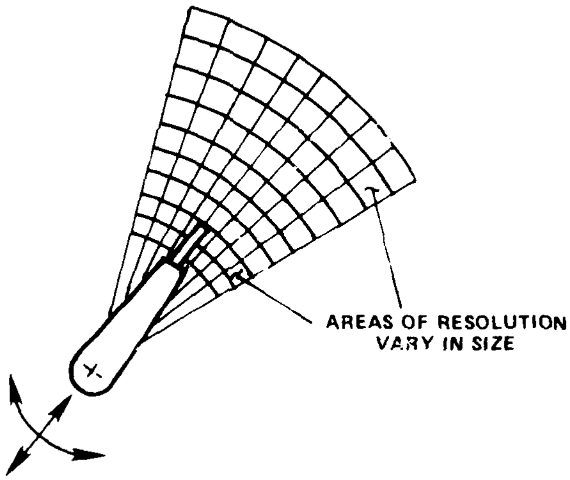


Figure 4-9 - Spatial Resolution of Revolute Robot

Accuracy, i.e., how close the robot end effector comes to a point to which it has not been previously, is a concept which applies only to robots with some off-line programming capability. For example, to say a robot's accuracy is 0.04" means that the tool tip can be positioned within a sphere of radius 0.04" centered on the programmed point P = (X,Y,Z). Accuracy is affected by the spatial resolution capability of the manipulator as well as by round-off errors in measurements and computations.

Repeatability refers to the precision with which the tool tip returns to a point to which it has been previously. For instance, if the robot is programmed to go to a point A by using a teach pendant, and if it has a repeatibility of 0.008", then when it is operating in automatic mode the tool tip will land within a sphere of radius 0.008" about A.

A manipulator has a natural structural frequency. This means that, if the arm is displaced from its position by some external force and then released, it will oscillate with this frequency. For many commercial robots this frequency is around 10 Hz. The simplest model of a manipulator link is a very stiff spring. In this case the structural frequency is given by the formula:

$$f = \frac{1}{2\pi}\sqrt{\frac{k}{M}}$$

where M is the manipulator mass and k is the spring constant.

For control purposes it is desirable that the structural frequency be small, so arms are built which are stiff (small value for k) and massive (large value for M). In practice the structural frequency is expressed in terms of inertia, since the inertia is what changes as the arm picks up and moves loads. If f_o is the structural frequency measured at some value of inertia J_o, then the frequency at some other value of J is

$$f = f_o\sqrt{\frac{J_o}{J}}$$

Generally the control system runs at about 60 to 80 Hz. If the control frequency and the structural frequency become sufficiently close, it is possible to produce oscillations in the robot. The wide variation of inertial and gravitational forces on the links and joints makes it a difficult problem to design an arm that will not break into oscillations in any operating range.

Oscillations can arise in several ways:

(1) The servo system design can allow the reaction of other joints to gravitational, inertial, or coriolis forces to exert forces on the joint under consideration and so to excite oscillations in certain operating ranges.

(2) Dropping a workpiece or a tool can set up vibrations which produce oscillations.

(3) Several arms working together can excite oscillations in one another, either through a commonly held workpiece or because of a common mounting.

Oscillations not only increase wear on the arm but also can increase the stopping time needed for the tool at some precise position (see Figure 4-10). In the worst case oscillations can cause the tool to hit another piece of equipment.

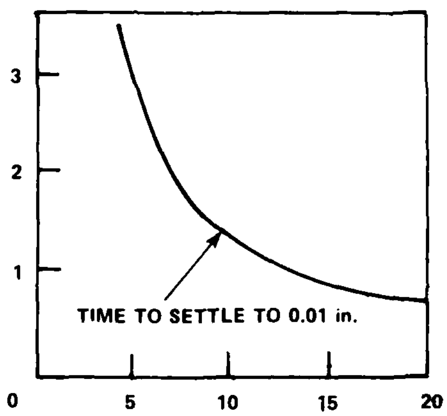


Figure 4-10 - Manipulator Settling Time

As can be seen from Figure 4-10, a high structural frequency is desirable to decrease settling time. However, a structural frequency close to the controller sampling frequency can be a cause of oscillation itself, and a design trade-off must be made.

5. CONTROL METHODS

In control theory a basic distinction is made between open-loop and closed-loop systems. In the robot classification scheme based on control, which was discussed in paragraph 1.3.2, non-servo robots use open-loop systems and servoed robots use closed-loop systems. The closed-loop systems involve some type of feedback mechanism (Figure 5-1).

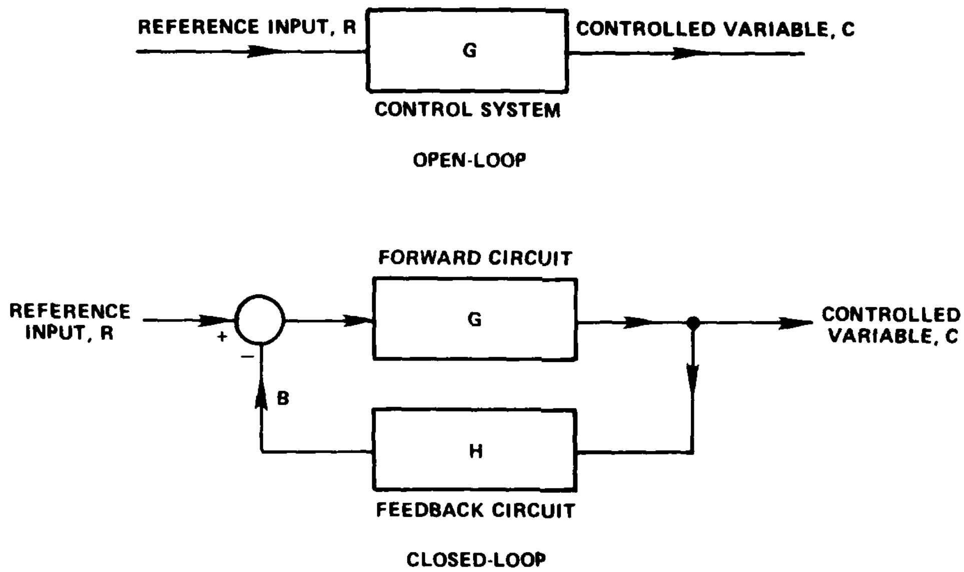


Figure 5-1 - Control Systems

An operating sequence for an open-loop system is described in paragraph 1.3.1. In such a system a signal is sent to a joint actuator and the joint then moves. No provision is made for comparing the position actually achieved with the desired position

Closed-loop control systems incorporate in the joint mechanism a measuring device, such as a potientiometer or encoder, which sends to the controller information about the position actually achieved. This position is compared with the desired position and a new signal proportional to the difference is sent to the actuator. The process is repeated until the measured error is less than some specified amount. A closed-loop system, or servo system, thus has three basic components: the controller-amplifier unit, a power unit (the motor driving the joint), and a measuring device. Control theory aspects of each of these components are discussed in this section.

5.1 POWER SOURCE

In electrically powered robots each joint is usually independently driven by a D.C. motor. The shaft of the motor is usually connected to some ball screw or gear mechanism which is directly connected to the link being driven. The quantities to be controlled are the position, velocity and the acceleration of the link. They are controlled by manipulating the current and voltage to the drive motor.

The response of a D.C. motor to given currents and voltages is described by the following three equations:[15]

$$V = L\,dI/dt + RI + K_E w \qquad (1)$$

$$T = K_T I \qquad (2)$$

$$T = J\,dw/dt + T_S \qquad (3)$$

where V is the voltage, I is the armature current, R is the motor resistance, K_E is a constant relating emf to motor speed, w is the shaft angular velocity, K_T is a constant relating current to torque, J is the motor inertia + load moment of inertia, and T_S is the load opposing torque + constant friction torque of the motor. These equations do not take into account viscous friction torques or other torques proportional to the shaft velocity.

These equations show that torque and position cannot be controlled independently. The two basic approaches to control are (a) to control torque by manipulating the current (Equation (2)), and (b) to control position by controlling the motor rotational speed through voltage manipulation (Equation (1)). However, the interdependence of V and I in these equations shows that there must be trade-offs between position and torque control. Position control is most important when picking up a piece, moving it, and at the very beginning of an assembly process. Torque control becomes important when two pieces are put together and excessive forces would result in damage to the parts. Koren and Ulsoy[16] have suggested a control strategy based on voltage manipulation with set limits on current. Figure 5-2[15] shows the torque-speed relationship for several types of electric motors. The simple curve for the D.C. motor allows it to be incorporated more easily into a servo-control system.

Similar control trade-offs are found in hydraulic systems.[17]

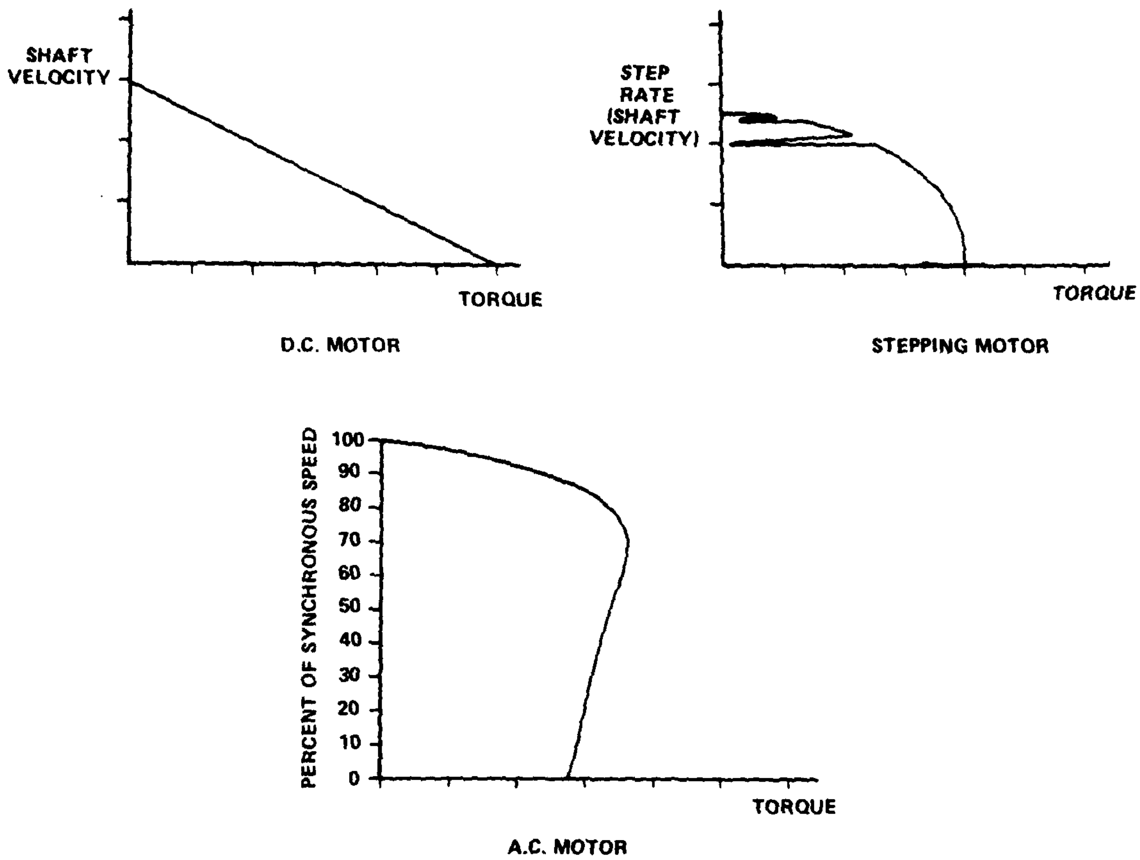


Figure 5-2 - Torque-Speed Relationship for Electric Motors

5.2 MEASURING DEVICES

The basic devices used to measure actual position or velocity in the feedback loops used in robots are potentiometers, encoders, resolvers, and tachometers. In addition, strain gauges are sometimes incorporated into the wrist structure to act as force sensors.

Potentiometers operate on the principle of a voltage divider in which the rotation of the motor shaft moves a wiper across a fixed resistor. This action results in an output voltage proportional to the angular position of the shaft,

$$V = K\theta$$

where V is the voltage, K is the constant of proportionality, and θ is the angle of rotation of the motor shaft. In a real potentiometer this relationship is not truly linear due to the presence of dirt, to wear of parts, and to the fact that Ohm's law is only a linear approximation. This means that the feedback system will have slightly different responses in different operating regimes.

Optical encoders can also be used to measure linear or rotational shaft motion and are becoming more common in robot control systems. Optical encoders typically have four basic parts: (1) a light source which is usually a light-emitting diode, (2) a light receiver which is usually a phototransistor, (3) a disc with a pattern of alternating translucent and opaque segments, (4) and some electrical device that converts sensor output (the response of the phototransistor) into a form that can be used by the robot controller (Figure 5-3).[15]

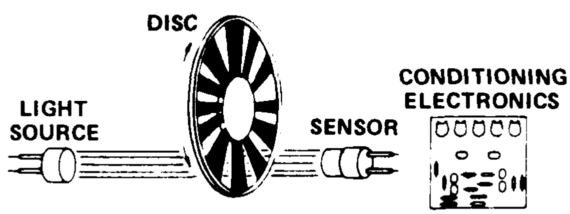


Figure 5-3 - Optical Encoder

As the disc rotates in response to shaft displacement, it sequentially blocks and transmits light to the receiver. The resulting light pattern indicates displacement and, when considered with respect to time, it indicates velocity or acceleration.

Encoders are of two types: incremental or absolute. An incremental encoder generates a pulse train by reading the number of increments traversed on the disc track as the encoder shaft rotates. The exact amount of travel is found by installing an auxiliary digital counter to count output pulses. Incremental encoders are used principally to measure velocity.

Absolute encoders generate a unique digital output for every shaft position. Thus positional information is read directly from the disc rather than from an external counter as with the incremental encoder. Consequently the information read from an absolute encoder is unaffected by power failure or noise.

Strain gauges are also used in feedback loops for torque or force control. Usually several are built into the wrist to sense forces and torques in all directions and about all axes (Figure 5-4).[18]

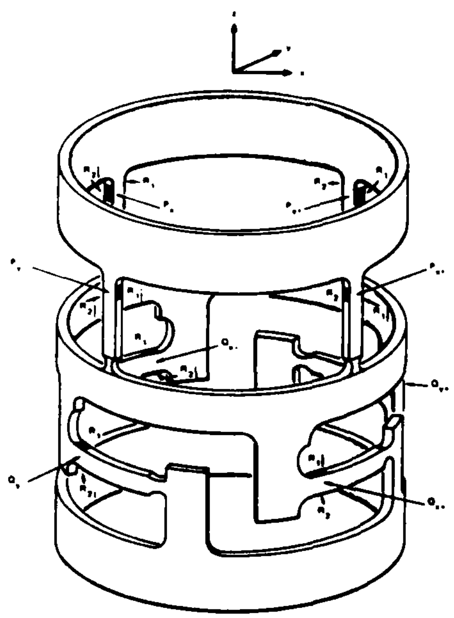


Figure 5-4 - Strain Gauge with Wrist Sensor

Although piezoelectric, magnetostrictive, and magnetic transducers can be used for the same purpose, strain gauges are the most commonly used since they are inexpensive, reliable, and rugged.

5.3 CONTROL ALGORITHMS

In present day servo-controlled robots, both point-to-point and continuous path, it is necessary to have a database specifying coordinates of the points to which the end effector is to move (see page 2.10). Points may be represented in Cartesian form or in terms of joint angles. The robot designer must provide an algorithm for calculating voltages and currents, or flow volumes and rates in hydraulic systems, to send to the joint actuators to achieve the desired position.

The controller contains either a data base or a computational method for generating the readings that an encoder or potentiometer would send when the position is achieved. The actual encoder readings are compared with this baseline, and the difference is called the error. Now the same calculations must be repeated so that signals can be sent to the actuators and they can compensate for the error. In a fully servoed system these computations might be done 80 times per second, so it is essential to have fast, efficient algorithms for these calculations. These operations are summarized in Figure 5-5.

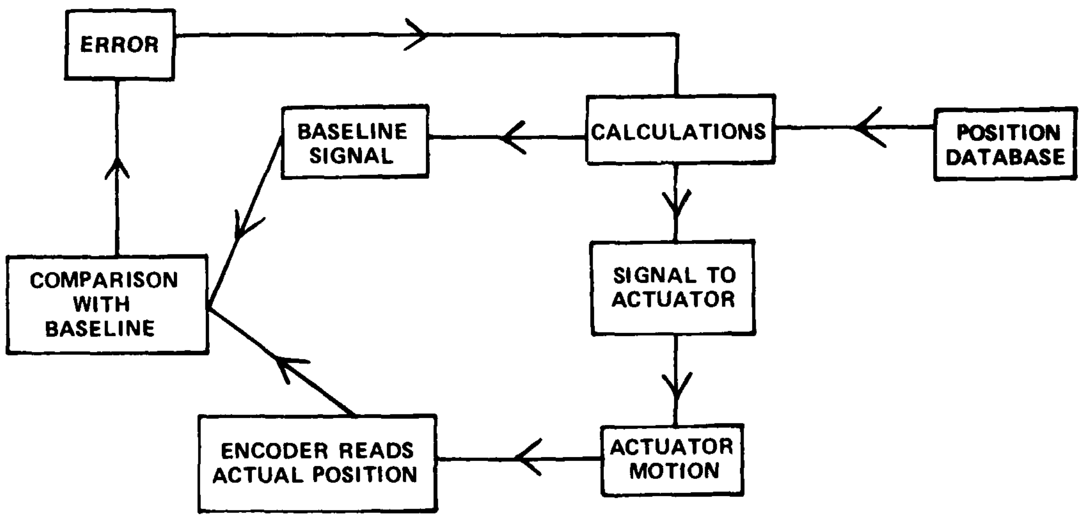


Figure 5-5 - Operations in a Robot Control System

The algorithms now in use are based on Lagrangian mechanics. The robot is viewed as a mechanical system with (usually) six degrees of freedom, one for each joint. In Figure 5-6 these degrees of freedom are indicated by the arrows.

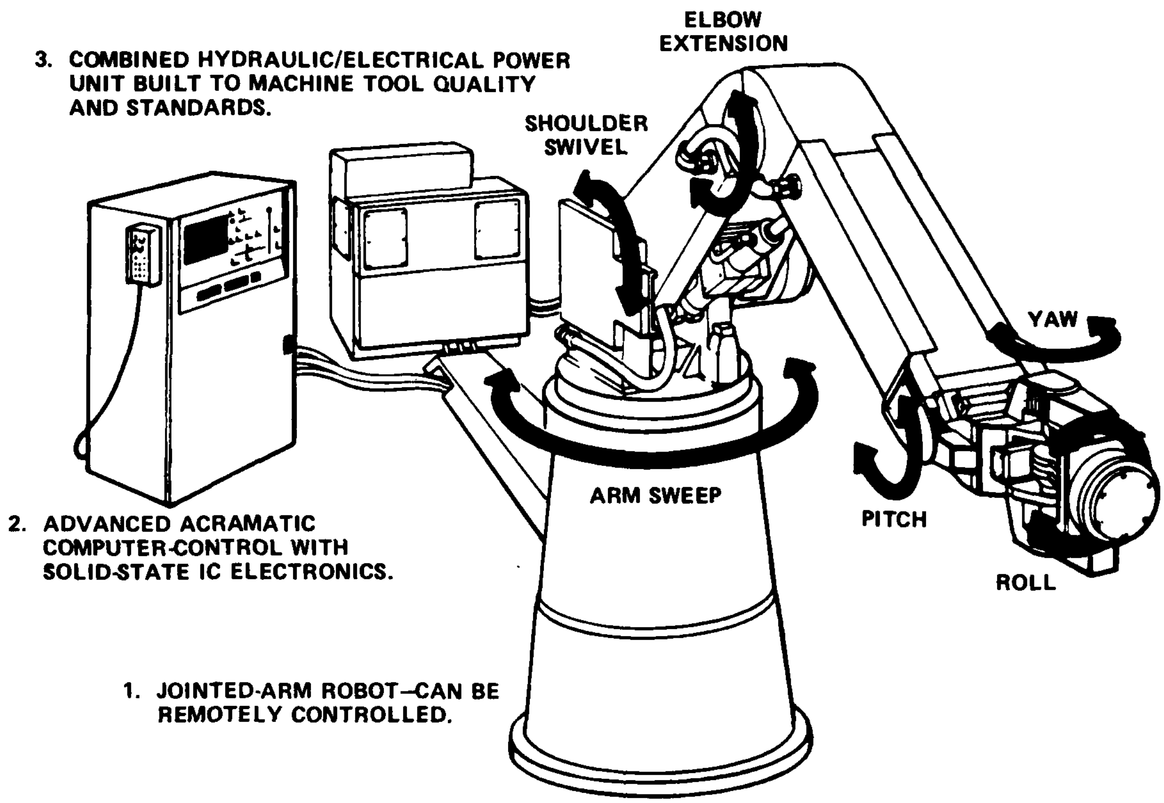


Figure 5-6 - Typical Manipulator Degrees of Freedom

Let $q(i)$ represent the position of the i^{th} joint, usually given in radians, and $\dot{q}(i) = dq(i)/dt$ the velocity of the i^{th} joint. Then the Lagrangian function, $L(q(1),\ldots,q(6),\dot{q}(1),\ldots,\dot{q}(6))$, is formed by subtracting the potential energy from the kinetic energy. Now the torque at the i^{th} joint, T_i , can be calculated by the following formula:

$$T_i = \frac{\partial L}{\partial q(i)} - \frac{d}{dt}\frac{\partial L}{\partial \dot{q}(i)} \tag{4}$$

The motion of the robot can be described reasonably well by this nonlinear system of six coupled second order differential equations.

$q(i)$ is given by the position data base or by the error signal. $q(i)$ and $q(i)$ are usually generated within the control system using fourth or fifth order spline functions. This means there is only nominal velocity and acceleration control. Given $q(i)$, $\dot{q}(i)$, and $\ddot{q}(i)$, T can be calculated using Equation (4). Then voltages and currents can be calculated using Equations (1), (2), and (3) on page 37.

If position control only, and not path control, is desired, this calculation may be done three or four times in the course of moving from point A to point B. This capability exists in many commercial manipulators now. If path control is desired, this calculation must be done 60 to 80 times per second. Until recently there were no algorithms fast enough to do this. The Newton-Euler method, developed by Luh, Walker, and Paul at Purdue[19,20], is a sequential computational technique for Lagrangian mechanics which is quite fast. Performing the calculation of T directly in Fortran takes about 7.9 seconds of computing time. With the Newton-Euler method in Fortran the computing time is 0.0335 seconds; in floating point assembly the computing time is 0.0045 seconds. Only three variations for specifying the path along which the robot is to move are currently provided in commercial controllers: (1) joints move in uncoordinated, unpredictable fashion to the desired end point, (2) joint variables are smoothly interpolated to final values in a coordinated fashion, or (3) the end effector moves in a straight line path toward the final destination. The PUMA 600 arm, for instance, incorporates both (2) and (3) in its control system. These approaches are explained in detail in "Robot Manipulators" by R.P. Paul.[21]

6. VISION AND TACTILE SENSORS

Most industrial robots in use today (1982) have no means for sensing the world about them. This lack of sensory input means that the parts which a robot handles must be precisely positioned, and the robot must be carefully trained or programmed before it can do a job. In terms of cost almost as much is spent on special tooling and jigs which enable the robot to do some assembly operations as on the robot itself.[22] A great deal of research effort is currently directed toward incorporating vision, tactile, force, and other contact and non-contact sensors into robotic systems.

Reasons for using external sensors on robots generally fall into two categories. First, they increase the flexibility of the robot, enabling it to be changed from one job to another more easily, since the need for complex set-ups is decreased. Second, some applications such as arc welding and parts sorting demand some type of sensing capability for effective utilization of the robot.

In this section vision and tactile sensors are discussed in some detail and other types of sensors are mentioned. Use of any fairly sophisticated sensor brings with it a set of control and data processing problems. These problems are discussed in this section and in Section 5.

6.1 VISION SYSTEMS

In a particular job a robot vision system is used to determine spatial relationships between tools and workpieces. This job encompasses tasks ranging from picking pieces out of a bin to positioning a welding torch. The robot vision system may also be used to inspect the tool and the workpiece. At present, because vision systems are at the leading edge of robotics technology, few such systems are being used in industrial settings.

In a broader sense vision systems should increase the flexibility of the robot. At present a robot's working environment must be highly structured, i.e., the pieces it works with must be carefully positioned. Figure 6-1[24] shows the set-up developed at Draper Labs to enable a robot to assemble an automobile alternator.

The assembly operation is impressive to watch. The number of special jigs, fixtures, and parts feeders required is also impressive. Designing and building such a setup is very costly, in terms of both time and money.

Moreover, the likelihood of problems with the operation is increased due to possible failures in the supporting equipment.

Vision systems should minimize the need for this peripheral equipment and should also allow robots to work more easily on moving lines. At present such an operation requires the use of some type of limit switch or timing device which enables the robot to calculate where the part is. The need for such devices means that the robot cannot be readily moved from line to line.

As attempts are made to use robotics in applications such as fire-fighting, military reconnaisance, and maintenance, the need for vision systems and other sophisticated sensors becomes more pressing. In these applications the environment in which the robot is to work is relatively unstructured. To get the job done the robot must have some way of knowing where things are and how they are changing.

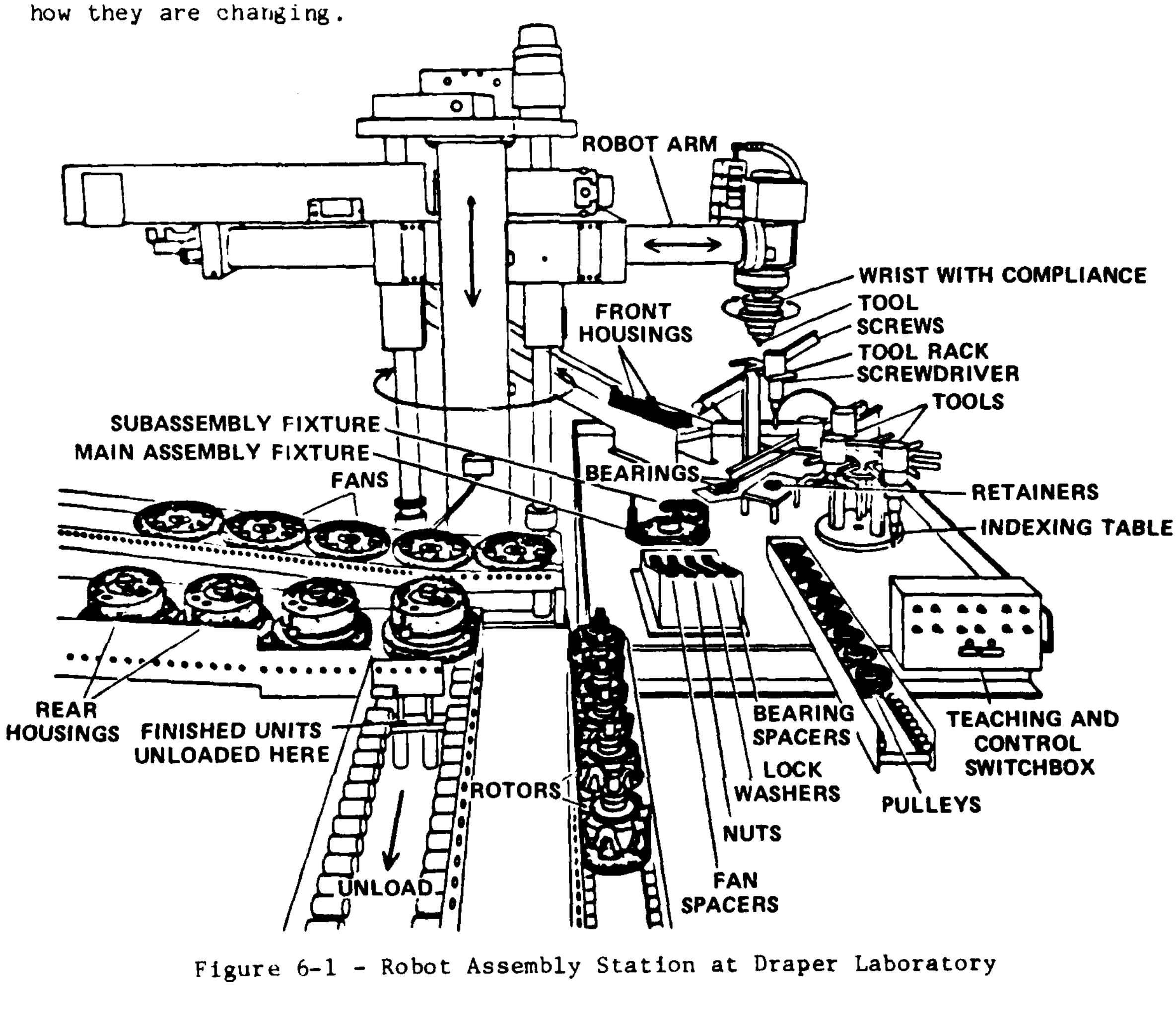


Figure 6-1 - Robot Assembly Station at Draper Laboratory

6.1.1 Applications

Machine vision can be applied in two broad areas of industrial processes: inspection of parts and tools, and sensor controlled manipulation. These areas of application are summarized in Figures 6-2 and 6-3.[23]

Robots with vision systems are in operation in only a few industrial settings. Current vision systems and those available in the near future will be limited to relatively controlled manufacturing environments or to very simple tasks in natural environments.

6.1.2 General Structure

A vision system has two basic components: a camera and an image processor. The relationship between these components and the robot control system is illustrated in Figure 6-4.[1]

In this illustration the DR-11 unit acts as an interpreter between the image processor and the robot control system. This step is necessary because the control system is not designed to interface directly with a complex sensor. At present Unimation and Automatix offer vision systems which are integrated with the controller. With other robots the user must patch the two systems together. Lechtman et al[25] give an account of the problems involved in such an operation.

The camera is usually a solid state device with the diode array ranging from linear to square. In the simplest systems the user sets a light intensity threshold level. When light of intensity greater than the threshold level hits the diode, a "1" is recorded for that position. If the light intensity is less a "0" is recorded. The resulting matrix of 0's and 1's is stored in the image processor and operated on using various algorithms to generate information about the scene. In more sophisticated systems the user can set a number of threshold levels to record a range of grays. Thus each diode might have 64 readings rather than just 2. In addition, subroutines can be written into the control and image processing program to change the threshold level according to changing circumstances.

The image processor takes the data generated by the camera and transforms it into information about holes, edges, position, or orientation. Thus the image processing operation consumes both computer time and memory. Current

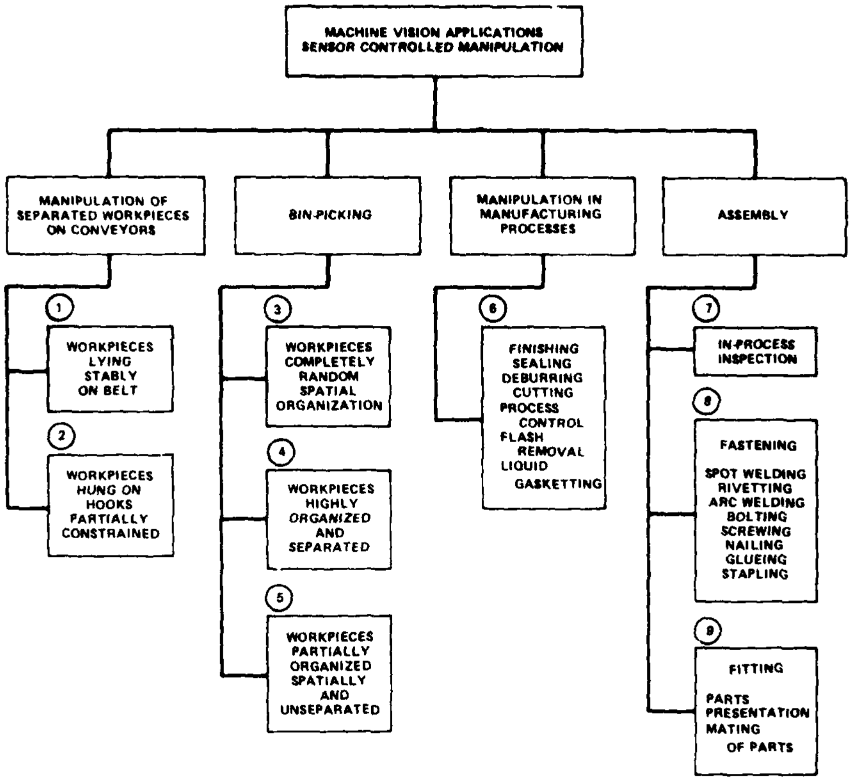


Figure 6-2 - Machine Vision - Sensor Controlled Manipulation

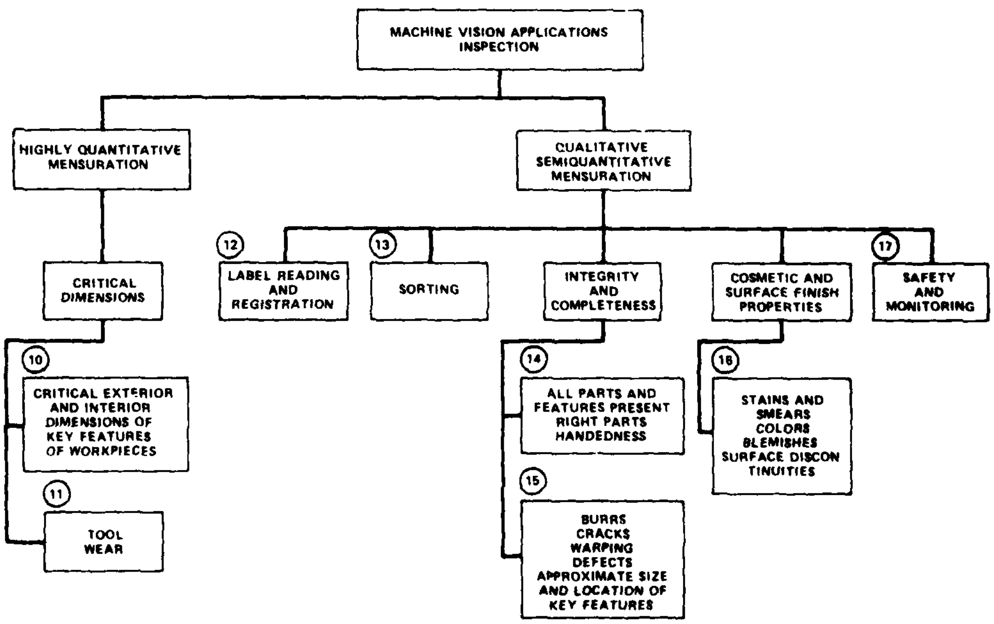


Figure 6-3 - Machine Vision - Inspection

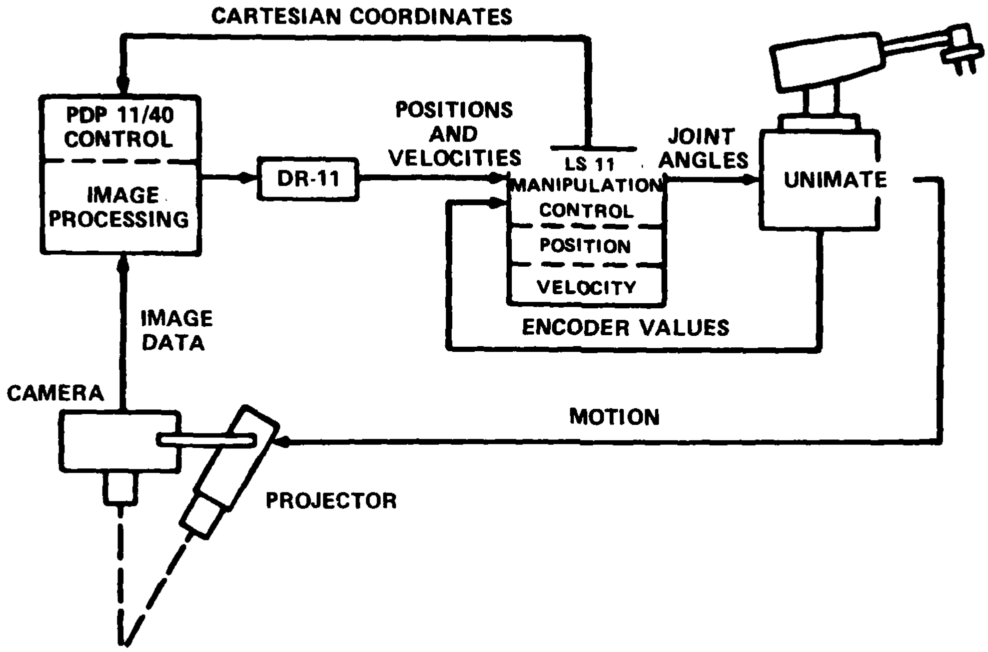


Figure 6-4 - Vision System - Robot Relationship

research is developing specialized chips and parallel processing techniques for image processing as well as faster processing algorithms.

The DR-11 interface device in the system illustrated in Figure 6-4 serves two purposes. One is to reformat the image processing information into a form acceptable to the robot controller; the other is to signal the robot controller that image information is available. The robot controller may be doing any of several tasks. When the interrupt signal from the interpreter arrives, the computer stops its current activity and immediately executes a predetermined subroutine.

The processing power needed for sensor systems in general is influenced by:

1. Image size
2. Response time
3. Computational complexity of the processing algorithm.

In vision systems the image size is the number of diodes in the camera. The diode array can range from a 128 x 1 array to a 256 x 256 array. Thus the image size can vary by a factor of 512.

Response time depends on the task at hand and on the robot controller. SRI has developed a vision system with a response time ranging from 100 to 500 milliseconds.[1] If the robot control system is working at a sampling rate of 60 Hz, then image information arrives about every 20 cycles, which leads to fairly smooth arm motions.

Reddy and Hon[26] estimate that a computational effort in the range of 1,000 to 10,000 operations per pixel is not unrealistic. This means that, for near real time image analysis, a machine must be able to handle about 1 to 100 billion operations per second.

Research on vision systems is proceeding on several fronts. The speed of visual processors is being improved through the use of special chips for edge-finding and fast fourier transforms, the use of parallel processing CPU's, and the development of VSLI devices. The physics of image formation is another area of research, involving understanding how the measurements obtained from the vision input device are determined by the lighting, shape, and surface material of the objects being imaged. Three-dimensional vision systems are being developed at several places. The basic problem in three-dimensional vision is to reconstruct 3-D information from the 2-D image recorded by the camera.

6.1.3 Examples

The two vision systems described briefly here are the CONSIGHT system developed at General Motors and a system developed at the National Bureau of Standards. Work on both systems was started in the mid to late 1970's. The CONSIGHT system was based on work done earlier at SRI and uses a camera and light source physically separated from the robot. Presently several companies offer robots with integrated vision systems.[9]

The CONSIGHT system (Figure 6-5) uses a camera and light source physically separated from the robot.[27]

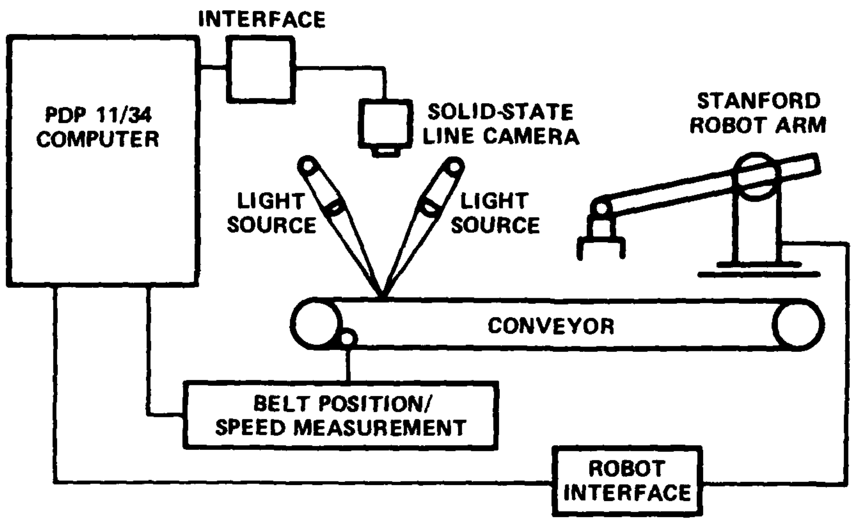


Figure 6-5 - CONSIGHT Vision System

The camera in this figure is solid state with a linear diode array so that it images a narrow strip across the conveyor belt. As the conveyor carries parts past the vision station, the part interrupts a sheet of light from one of the two light sources and casts a shadow on the line being watched by the camera, thus locating the part for the robot. Two light sources are needed because, with only one source, a tall piece would cast a shadow on the line of sight of the camera long before the piece actually reached that position. Using two sources and appropriately adapting the angle of incidence of the light beams eliminates this shadow effect.

The CONSIGHT system was designed from the outset for an industrial environment. It is easy to set up and calibrate. Calibration takes about 15 minutes. A simple "teach-by-show" routine enables the system to recognize new parts. The system does not depend on high contrast lighting. Finally, the system is modular, i.e., it is partitioned into robot, vision, and monitor subsystems, any one of which can be changed with minimal impact on the others.

The robot vision system developed at the National Bureau of Standards[27] has a configuration which differs markedly from the CONSIGHT system. Both the camera and the light source are mounted on the arm. The camera is solid state with a 128 x 128 diode array. The light is a stroboscopic light which emits a plane of light through a cylindrical lens (Figure 6-6).[28]

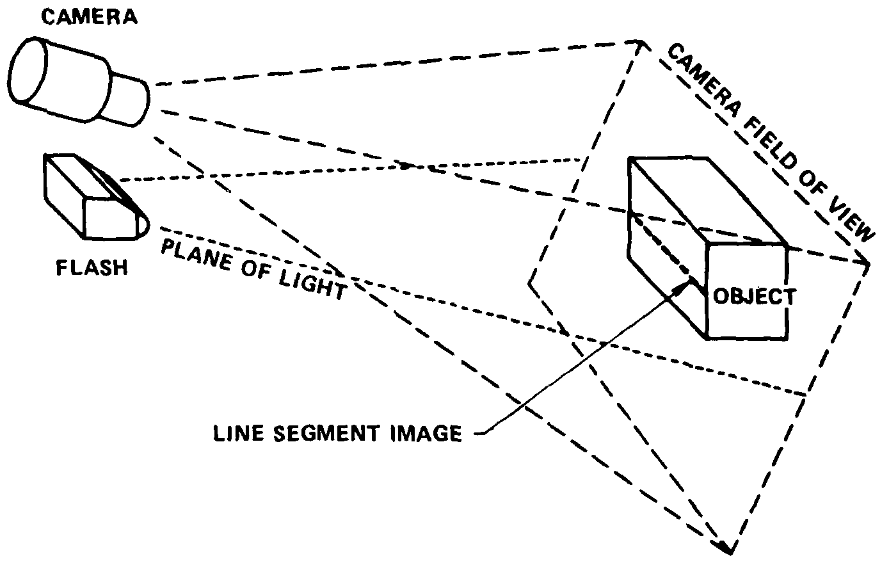


Figure 6-6 - NBS Vision System

The effect of this arrangement is to ensure that each column of the image has at most one intersection with the plane of light. Thus when the robot "looks" at an object such as the cylinder in Figure 6-7,[28] the camera records a line.

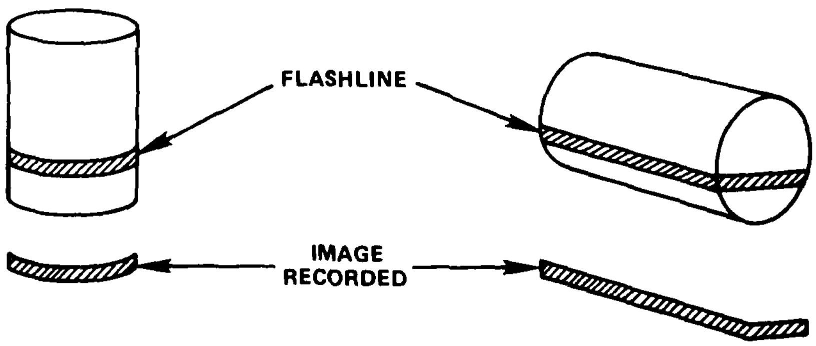


Figure 6-7 - Images of a Cylinder Using NBS System

The *pattern recognition* system takes the pattern of lines and generates information about the object.

The NBS system has been used in experiments in the sorting and acquisition of simple parts. More recently it has been developed to the point where the arm can follow and pick up a slowly rolling cylinder.

Noncontact sensors, other than vision systems, can be useful for object avoidance and tool or parts inspection. Such sensors can be based on laser ranging, radar-like acoustic devices, eddy current effects, or thermal sensing.

6.2 TOUCH SENSORS

Tactile sensing systems can be used to sense force, proximity, texture, and shape. Of these, force and proximity are directly sensed using devices such as strain gauges and microswitches; texture and shape are inferred from force and proximity data using pattern recognition methods. Touch sensors can be used to identify parts, to follow edges for welding or deburring operations, or for assembly operations. ASEA has developed a system in which a robot operating a deburring grinder is able to follow the edge of the casting by means of a sense of touch. Unlike robots that must have the path precisely specified in advance, the ASEA robot automatically adjusts to variations in the position or shape of the workpiece. Other industrial operations in which a tactile sensor might be used are gauging, qualitative inspection, painting, sorting, polishing, casting, forging, and pick-and-place operations.

These applications generate some stringent requirements that touch sensors must meet before they can be used in an industrial setting. Sensors must be small enough to fit into grippers and accurate enough to avoid crushing or dropping pieces. And they must be rugged enough to remain accurate through repeated use and mechanical abuse such as minor collisions. Specialized applications may require special materials for durability to survive heat, hard radiation, salt water, high pressure, electrical interference, or other hazards.

In the Artificial Intelligence Lab at MIT a high resolution touch sensor has been developed.[29] The device is an array of 256 tactile sensors which fits on the tip of a finger. Each sensor covers an area of about .01 sq cm and gives an analog indication of the force on its surface over a range of 1-100 grams. This finger-tip size sensor has been used to distinguish among some commonly used fasteners such as nuts, bolts, flat washers, etc., by touching them and then analyzing the tactile image. The sensor is constructed from conductive silicone and printed circuit boards (Figure 6-8).[29]

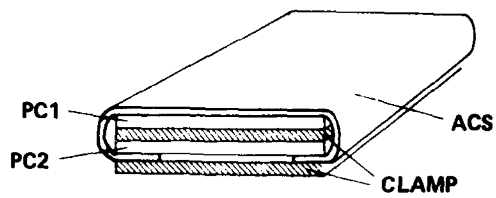


Figure 6-8 - MIT Touch Sensor

A similar sensor, which is a sandwich of conductive rubber, glass insulator, metal, and integrated circuit, has been developed at Carnegie-Mellon and the California Institute of Technology.[30] The special feature of this sensor is a VLSI computing circuit which is part of the sensor itself. This circuit transduces the pattern of applied force, reduces the data with parallel processing, and multiplexes data for communication over a compact serial channel.

At NASA's Jet Propulsion Lab a pressure sensor which uses optical fibers has been devised (Figure 6-9).[31] The fibers bring light into cells on the gripping surface. The light is then reflected from a flexible covering onto fibers leading to detectors. Distortion due to tactile pressure changes the amount of light reflected.

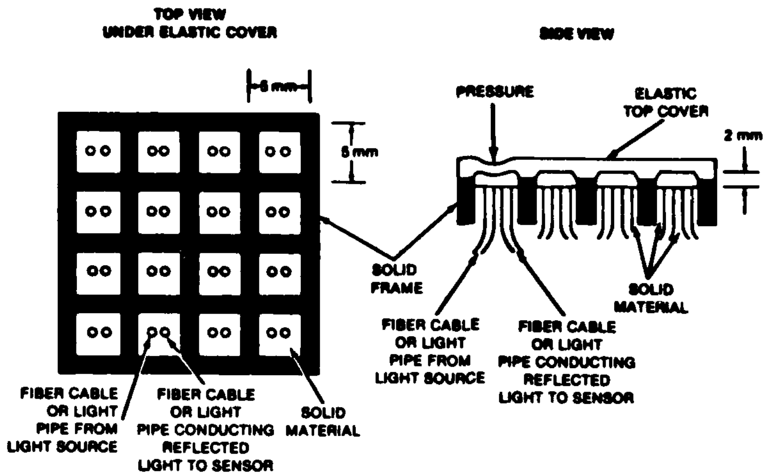


Figure 6-9 - NASA Optical Tactile Sensor

Optical fibers have been incorporated into a robot tactile sensor system by Souriau et CIE of France.[30] Their system, however, is designed only to locate parts, not to sense pressure or other tactile information. Optical fibers are used in these sensors because they make the system immune to certain environmental disturbances such as high temperature and electrical noise. Electrical noise in industrial enviroments can be a problem for these sensors because of the low signal levels associated with low contact pressure.

The pattern recognition problem for tactile sensors should be much more tractible than for vision sensors. There are far fewer data to be analyzed than for a vision system. The collection of data is more controlled in that there is less background noise. Analyzing a tactile image is like analyzing a visual image with controlled backgound, illumination, and point of view. Moreover, the properties of interest in tactile sensing (pressure, texture, shape) are closer to the properties actually measured by the sensor. In vision systems mechanical properties must be inferred from optical properties.

7. END EFFECTORS

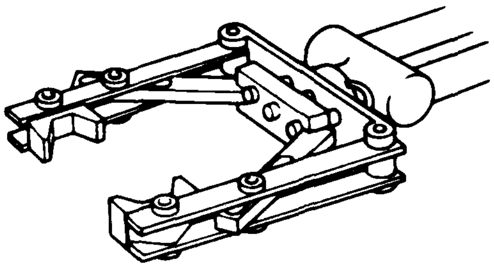
An end effector is the hand of the robot. It might be a multipurpose gripper as in Figure 7-1[4] or a very specialized tool as in Figure 7-2.[4] The end effectors must be rugged enough to withstand accidental collisions and other hazards of the workplace. If they are too heavy, they slow the manipulator and significantly reduce the load the manipulator can carry. To reduce the amount of time spent in changing tools during a job, the end effector ideally should perform several different functions. In many applications the user must design and have built the end effectors needed for a particular job. Most manufacturers of robots offer only a limited line of such devices, and the lack of standardized interfaces has inhibited the development of companies specializing in them. To date, grippers have almost always had to be customized and in consequence have accounted for roughly 20-30% of the cost of a manipulator. For some jobs, such as assembly tasks, the robot might use an array of end effectors which it changes itself. The robot assembly station built at Draper Labs to assemble an automobile generator works this way (see Figure 6-1, Section 6).

7.1 TYPES OF END EFFECTORS

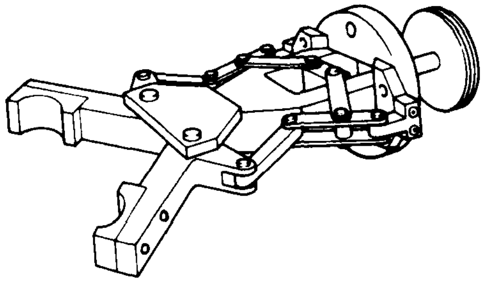
Of the several ways to characterize end effectors, two are presented here: (1) by grip mode, and (2) according to use. The four basic types of grip are illustrated in Figure 7-3.

A detailed discussion of the various types of vacuum and magnetic grippers is given by Engleberger.[4] All of these types of grippers usually need self-aligning jaws to ensure that parts are contacted evenly at two spots. It should be fairly clear that the nature of the job determines the appropriate manner of gripping for the end effector. Moving thin sheets of metal will require vacuum cups or magnets; loading forgings into a machine tool will require large steel grippers.

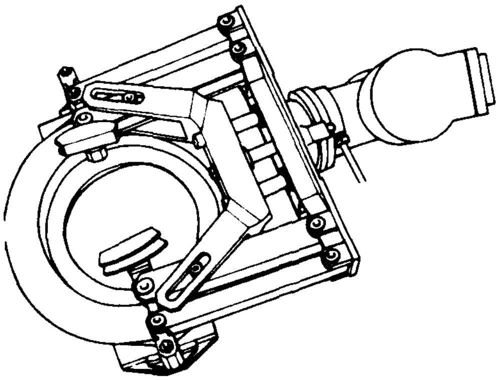
End effectors can also be classified by their range of use. Job specific end effectors are those which incorporate a tool for performing a specific jobs such as a welding torch or a paint sprayer (see Figure 7-2). General purpose grippers can carry a range of tools or objects (see Figure 7-1).



CAM-OPERATED HAND

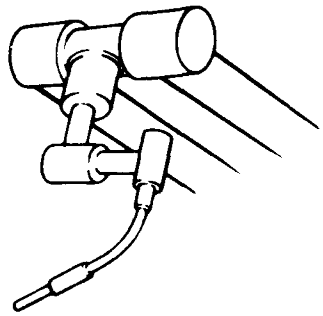


WIDE-OPENING HAND

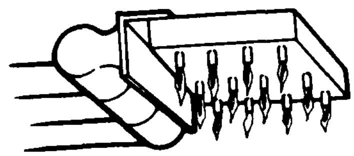


CAM-OPERATED HAND WITH
INSIDE AND OUTSIDE JAWS

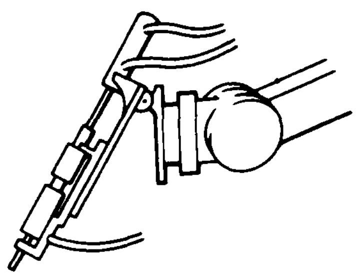
Figure 7-1 - Multipurpose Grippers



ARC WELDING TORCH

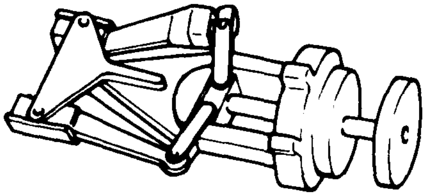


HEATING TORCH

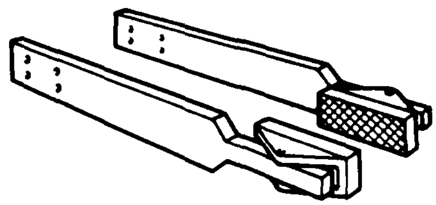


STUD WELDING HEAD

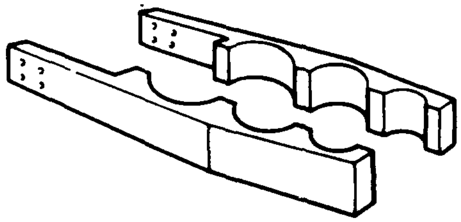
Figure 7-2 - Specialized Grippers



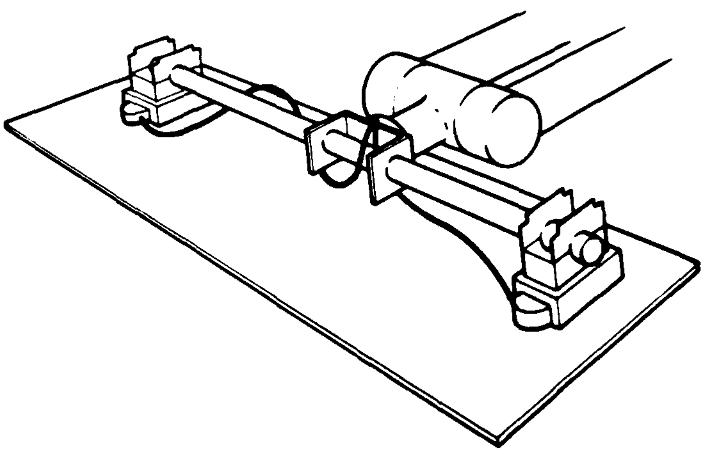
STANDARD HAND



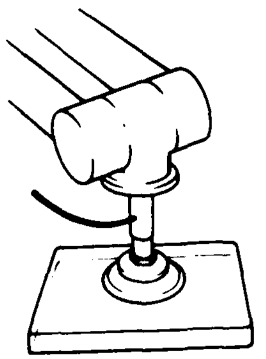
FINGERS SELF-ALIGNING



FINGERS FOR GRASPING DIFFERENT
SIZE . ARTS



MAGNETIC



SIMPLE VACUUM CUP HAND

Figure 7-3 - Grip Mode

Recent research at the University of Birmingham in England has attempted to improve the performance of linkage-type grippers. The design which has been developed (Figure 7-4) incorporates a pair of special cams which control the position of the gripper fingers so that cylindrical components of different diameters can be clamped without the problem of offset and subsequent inaccurate placement of the workpiece.

Current end effector/manipulator systems have two basic problems: (1) they cannot adapt to a wide range of object shapes, and (2) they cannot make small displacements at the hand without moving the entire arm. This limits the response and fidelity of force and position for hand movements to that of the entire arm, even for very small motions.

7.2 GRIPPER CONTROL

Once an end effector has been chosen for a specific set of jobs, certain relations between the gripper control and the task characteristics must be considered. The gripper holds the object to prevent translation or rotation relative to the gripper. As noted earlier there are three basic methods for doing this: friction between the part and the gripper, physical constraints which range from fingers to a ladle, and attraction by magnetic or suction devices. Suppose part of a task involves using a gripper, such as that in Figure 7-4, to move a 30-lb cylindrical forging from a pallet to a milling machine. A robot at normal full speed can, during acceleration and deceleration, exert forces on the part of 2g (twice the earth's gravity). If the part is moved with its long axis vertical, only the friction between the part and

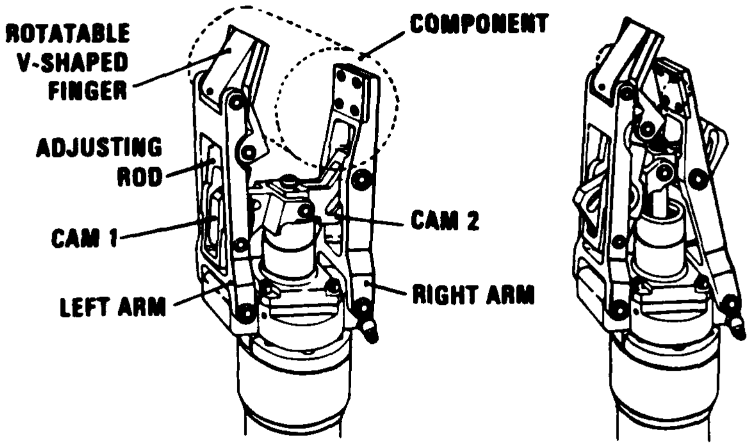


Figure 7-4 - Concentric Linkage Gripper

the gripper keeps the part from falling. The gripping force which must be applied to prevent the part from slipping can be found through the following computation:

$$\text{gripping force} \times \text{coefficient of friction} = \text{weight} \times \text{g-load} \qquad (1)$$

The g-load in this case will be 3 (1 for the earth's gravity and 2 from robot acceleration), and a plausible value for the coefficient of friction is 0.15. Thus

$$\begin{aligned} \text{gripping force} &= (30 \times 3)/\ 0.15 \qquad (2) \\ &= 600 \text{ lb} \end{aligned}$$

Allowing a safety factor of two makes the gripping force 1200 lb. If the part is carried with its long axis horizontal, slippage will be caused only by accelerations of the robot, not by gravity, so in Equation (2) the g-load factor is two rather than three. Thus with the piece carried horizontally and a safety factor of two, the gripping force would be 400 lb. Similar calculations can be made for any of the other gripper modes used. One area of current research is the development of gripper pads with very high coefficients of friction.

7.3 SENSORS IN END EFFECTORS

The simplest sensors built into end effectors are limit switches or electric eyes which indicate only whether the gripper has contacted something or whether something is between the gripper fingers. Figure 7-5[30] shows a gripper with a solid state light-emitting diode in one gripper plate, which acts as a transmitter of light, and a solid state photodiode in the other plate, which acts as a receiver.

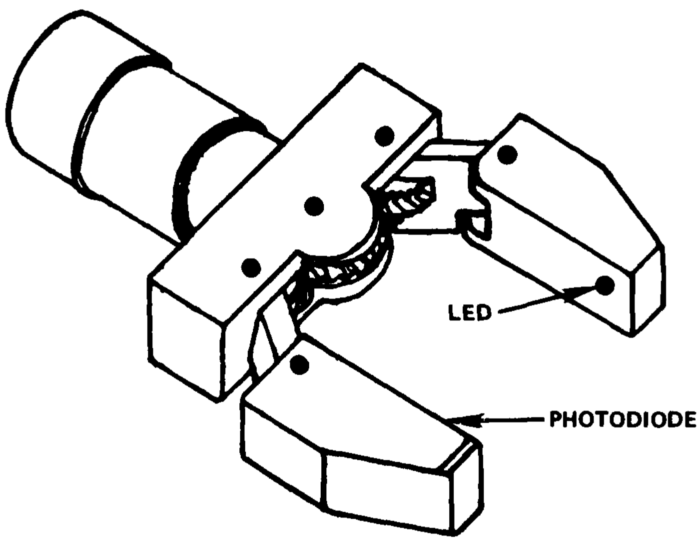


Figure 7-5 - Gripper with Built-In LED-Based Sensor

As robot technology has developed, applications in assembly and inspection have emerged as well as more complex pick and place operations. These new applications have generated a demand for "smart" end effectors which incorporate more sophisticated sensors than those just described.

End effector sensors being experimented with range from various types of touch pads to air jets and miniature cameras. The French company Souriau et CIE has developed a multipoint detection system using fiber optics. The system uses four detectors, one in each of three fingers and one in the "palm" of the gripper mechanism. Each detector uses two optical fibers, one to emit light and one to detect it. The information is used to position the fingers so that all three are in contact with the workpiece and ready to grip it. One advantage to using optical fibers is that they make the system immune to environmental disturbances such as very high temperatures. A number of companies have developed fiber optic sensors specialized to welding applications.

Edel and Jolly at the Universite des Sciences et Techniques de Lille in France have tested a three-fingered hand which uses air jet sensors located in the fingers to detect boundaries of workpieces. Each finger has six air jets and air pressure sensors. The pressure measurements are converted to distances by the control system, allowing fingers to track boundaries without actually touching them.

Another type of non-contact sensing being tried involves mounting cameras directly in the gripper (Figure 7-6). Such a system has been developed and attached to a PUMA robot. This "eye-in-hand" system can distinguish among several different parts and carry out simple inspections.

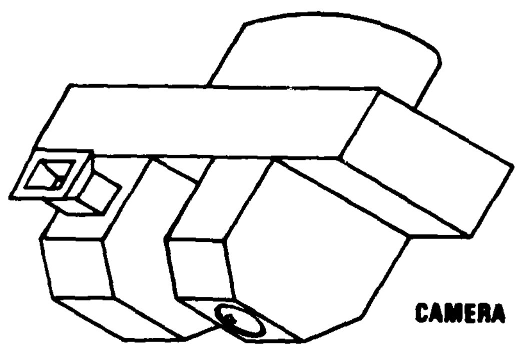


Figure 7-6 - Gripper Mounted Camera

Direct contact touch sensors are rapidly becoming more sophisticated and reliable as both sensor technology and image processing techniques are developed. A 1980 survey of industry and researcher views on tactile sensors indicated that touch sensing was seen as an essential concomitant of vision systems.[22]

At MIT a touch pad roughly the size of a fingertip has been devised. This pad contains an array of 256 sensors, giving it a resolution of .01 sq cm and an analog indication of the force on its surface over a range of 1-100 grams. This touch system has been used to distinguish among nuts, bolts, flat washers, cotter pins, and several other common fasteners.[29]

Joint work at the California Institute of Technology and Carnegie-Mellon University has led to a tactile sensor which incorporates a VLSI computing circuit as a part of the sensor. The sensor itself is a sandwich of conductive rubber, glass insulator, metal, and the integrated circuit device.

7.4 THREE-FINGERED GRIPPERS

The human hand is an exceptionally versatile end effector. We can hold and operate heavy tools as well as thread a nut onto a bolt or pick up small ball bearings. The various research efforts focused on developing three-fingered grippers are aimed at devising a robot end effector with a comparable range of flexibility. This research involves not only mechanical design of the gripper but also a kinematic analysis of grasping to isolate those factors crucial to manipulating and securing an object and development of software systems to control the gripper.

The first serious effort in this direction seems to have been by Skinner in the mid 1970's. The Skinner hand is a three-fingered design in which each finger is a prehension, and a joint at the base of each finger allows it to twist about its long axis (Figure 7-7).[32] This gripper could be used for friction, physical constraint, and suport modes of gripping.

More recently Salisbury at Stanford has built a three-fingered hand with nine degrees of freedom, each finger having three degrees of freedom.[33] The hand is powered by twelve small electric motors mounted on the forearm of a PUMA robot. The detailed design of this hand stemmed from a careful analysis of grasping and manipulation which allowed optimization techniques to be applied to such things as link dimensions.

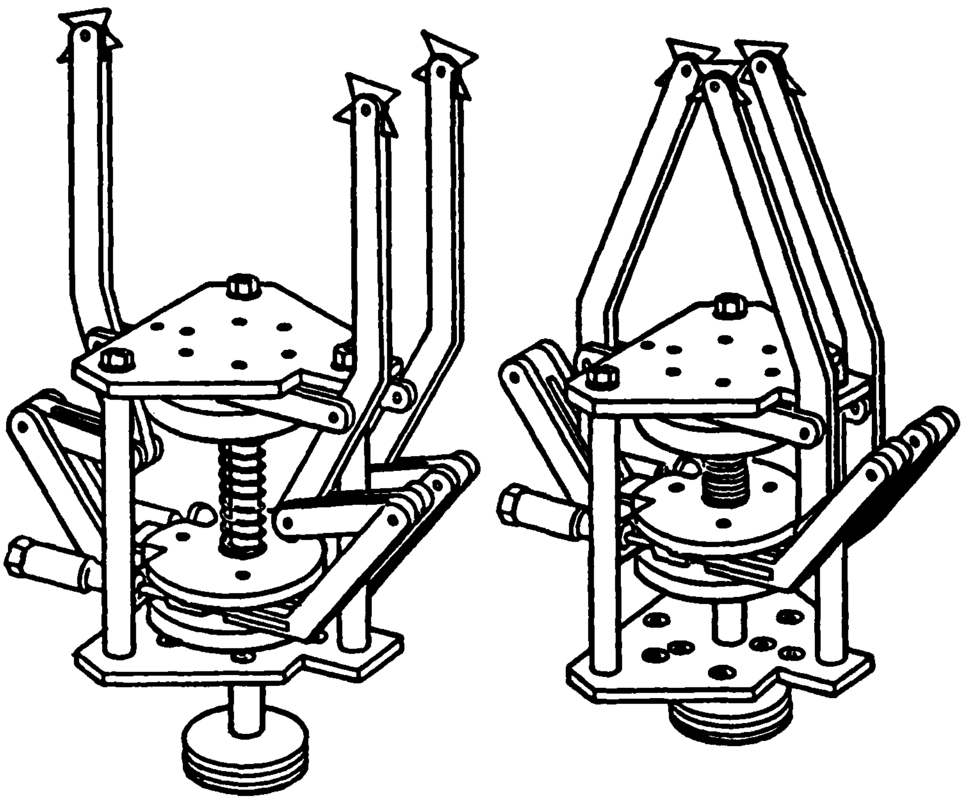


Figure 7-7 - The Skinner Hand

Okada has also developed a hand with three multijointed fingers. The time consuming aspect of developing such grippers is software development. Figure 7-8[34] shows results of Okada's three-fingered hand attaching a nut to a bolt.

7.5 REMOTE CENTER COMPLIANCE DEVICES

In many assembly operations there is very little margin for error. Pieces may fit together with a tolerance of 0.001 in. Attempting to have a robot with a repeatability of 0.004 in. perform such a task will clearly lead to problems. At Draper Labs a robot-end effector interface called a remote center compliance tool was invented in the mid 1970's to enhance the precision of robot manipulations. The tool allows an object to comply in reponse to forces arising from contact with the edges of a hole. The mechanical working of this device is explained in Figures 7-9, 7-10, and 7-11.[24] The remote center compliance tool is termed a passive compliance device because it uses no feedback loop.

7.6 ROBOT-END EFFECTOR INTERFACES

The physical connection between the robot manipulator and the end effector is an important design consideration. The interface must support the end effector structurally, so it must be strong enough to withstand the inertial forces arising from rapid acceleration of full loads. If some degree of compliance is desired, the interface might include a remote center compliance device. For safety reasons some mechanical device or or sensor should indicate when the wrist is over-loaded.

Both power to the gripper and information from it must past through the wrist. The design of the gripper must provide for electrical, hydraulic, pneumatic, or mechanical connections to pass power to the gripper. Of these connections the most difficult to implement reliably is that for hydraulic power because hydraulic fluid is so easily contaminated when end effectors are changed. Such contamination can cause a servo valve to stick and result in rapid, unpredictable manipulator motion.

Information transfer will become more important as more sensors are placed on the end effectors. Since signals from gripper sensors to the control computer are usually transmitted at low power levels, connector design is fairly straightforeward. The low power level, however, makes for a high signal to noise ratio which can cause processing problems. The principal means for signal transfer are through electrical, pneumatic, or fiber optic connections.

Figure 7-8 - Three-Fingered Gripper Threading a Nut onto a Bolt

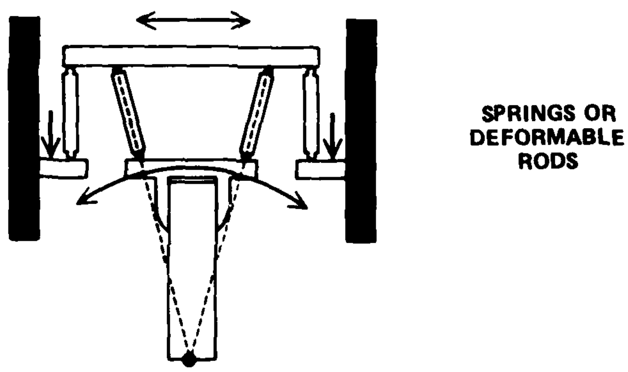


Figure 7-9 - Remote Center Compliance Device

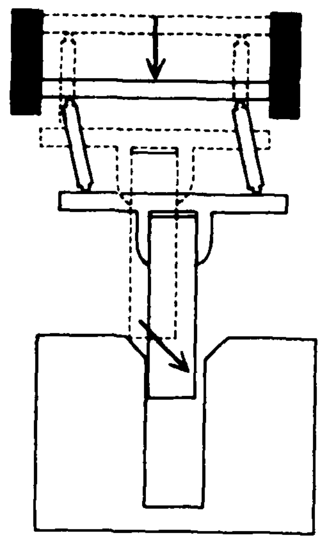


Figure 7-10 - Vertical Links Compensate for Lateral Misalignment

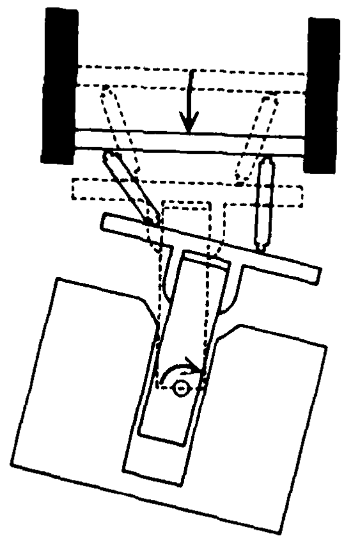


Figure 7-11 - Slanted Links Compensate for Rotational Misalignment

REFERENCES

1. Agin, G.J., "Real Time Control of a Robot with a Mobile Camera", Stanford Research Institute Technical Note 179, February 1979.

2. "Innovation and Transfer of U.S. Air Force Manufacturing Technology", Committee on Computer-Aided Manufacturing, National Research Council, National Academy Press, Washington, D.C., 1981.

3. Tanner, W.R., "Basics of Robotics", Industrial Robots, Vol. 1: Fundamentals, Society of Manufacturing Engineers, Dearborn, Michigan, 1981.

4. Engleberger, J., "Robotics in Practice", American Management Association, 1980.

5. Yonemoto, K., "The Art of Industrial Robots in Japan", Proceedings of the 11th International Symposium on Industrial Robots, Tokyo 1981.

6. Ardayfio, D.D., and H.J. Pottinger, "Computer Control of Robotic Manipulators", Mechanical Engineering, Vol. 104, No. 8, Aug. 1982.

7. Kempf, K.G., "Robot Command Languages and Artificial Intelligence", Robots VI, Conference Proceedings, Society of Manufacturing Engineers, Dearborn, Michigan, 1982.

8. Girod, G.F., "Utilization of Microprocessors in Robot Automation", Robots VI, Conference Proceedings, Society of Manufacturing Engineers, Dearborn, Michigan, 1982.

9. 1982 Robotics Industry Directory, Technical Data Base Corp., 1982.

10. Doty, K.L., "Intelligent Robots: the Integration of Microcomputer and Robotic Technology", Course notes, George Washington University, July 1982.

11. Yeaple, F., "Hydrostatic Motors Applications Guide", Design Engineering, January 1982.

12. "German Robot Industry Set To Take On The Opposition", The Industrial Robot, March 1982.

13. "Robots: Lengthening Man's Reach", Design Engineering, January 1982.

14. Tsai, Y.C. and A.H. Soni, "Accessible Region and Synthesis of Robot Arms", Journal of Mechanical Design, Vol. 103, October 1981.

15. "D.C. Motors, Speed Control, and Servo Systems", Electro-Craft Corporation, 5th Ed., 1980.

16. Koren, Y., and A.G. Ulsoy, "Control of D.C. Servo-Motor Driven Robots", Robots VI: Conference Proceedings, Society of Manufacturing Engineers, Dearborn, Michigan, 1982.

17. Dransfield, P., "Hydraulic Control Systems - Design and Analysis of Their Dynamics", Springer-Verlag, 1981.

18. Rosen, C.A., and D. Nitzan, "Use of Sensors in Programmable Automation", Industrial Robots, Vol. I: Fundamentals, ed. Tanner, Society of Manufacturing Engineers, 1981.

19. Luh, J.Y.S., M.W. Walker, and R.P. Paul, "On-Line Computational Scheme for Mechanical Manipulators", Journal of Dynamic Systems, Measurement, and Control, Vol. 103, June 1980.

20. Silver, W.M., "On the Equivalence of Lagrangien and Newton-Euler Dynamics for Manipulators", The International Journal of Robotics Research, Vol. 1, No. 2, Summer 1982.

21. Paul, R.P., "Robot Manipulators", MIT Press, Cambridge, Mass., 1982.

22. Harmon, L.D., "Automated Tactile Sensing", MSR 82-02, Society of Manufacturing Engineers, Dearborn, Michigan, 1982.

23. Rosen, C.A., "Machine Vision and Robotics: Industrial Requirements", Computer Vision and Sensor-Based Robots, Dodd and Rossel (eds.) Plenum Press, New York and London, 1979.

24. Whitney, D.E. and J.L. Nevins, "Computer-Controlled Assembly", Scientific American, February, 1978.

25. Lechtman et al, "Connecting the PUMA Robot with the MIC Vision System and Other Sensors", Robots VI: Conference Proceedings, Society of Manufacturing Engineers, Dearborn, Michigan, 1982.

26. Reddy, D.R. and R.W. Hon, "Computer Architectures for Vision", Computer Vision and Sensor-Based Robots, Dodd and Rossel (eds.), Plenum Press, New York and London, 1979.

27. Ward, M.R. et al, "CONSIGHT: An Adaptive Robot with Vision", Robotics Today, Summer 1979.

28. Nagel, R.N. et al, "Experiments in Part Acquisition Using Robot Vision," Robotics Today, Winter 1980-81.

29. Hillis, W.D., "A High Resolution Image Touch Sensor", The International Journal of Robotics Research, Vol 1, No. 2, Summer 1982.

30. "A Report on the 12th ISIR in Paris", Robotics Today, Vol. 4, No. 4, August 1982.

31. "Touch Sensor Responds to Contact Pressure", NASA Tech Briefs, Summer 1981.

32. Skinner, F., "Design of a Multiple Prehension Manipulator System", ASME Pub. No. 74-DET-25, October 1974.

33. Salisbury, J.K. and J.J. Craig, "Articulated Hands: Force Control and Kinematic Issues", paper presented at 1981 Joint Automatic Control Conference, June 1981, Univ. of Virginia, Charlottesville, Virginia.

34. Okada, T., "Computer Control of Multijointed Finger System for Precise Object-Handling", IEEE Transactions on Systems, Man, and Cybernetics, Vol. SMC-12, No. 3, May/June 1982.

INITIAL DISTRIBUTION

Copies	
5	NAVAIR/Code 034C
12	DTIC

CENTER DISTRIBUTION

Copies	Code	Name
1	1800	.H. Gleissner
2	1809	D. Harris
1	1822	A. Camara
1	1824	S. Berkowitz
1	1824	G. Marques
1	185	T. Corin
1	185	R. Jenkins
1	187	M. Zubkoff
15	1873	R. Johnson
1	1873	S. Ullman
1	1873	R. Putnam
1	2821	A. Pollack
10	5211.1	Reports Distribution
1	522.1	Unclassified Lib (C)
1	522.2	Unclassified Lib (A)
1	93	L. Marsh

DATE

FILMED

8